Universitat de les Illes Balears

MASTER'S THESIS

MACHINE LEARNING FOR REMOTE SENSING OF *XYLELLA FASTIDIOSA*

Javier Galván Fraile

Master's Degree in Physics of Complex Systems

(Specialization/Pathway in Complex Systems)

Centre for Postgraduate Studies

Academic Year 2019/2020

MACHINE LEARNING FOR REMOTE SENSING OF *XYLELLA FASTIDIOSA*

Javier Galván Fraile

Master's Thesis

Centre for Postgraduate Studies

University of the Balearic Islands

Academic Year 2019/2020

Keywords:

Machine learning, *Xylella fastidiosa*, Remote sensing, Satellite imagery, WorldView2, Artificial Neural Networks, Recurrent Neural Networks

Thesis Supervisor's Name: Dr. José Javier Ramasco Sukia

Thesis Supervisor's Name: Dr. Manuel Alberto Matías Muriel

“Our intelligence is what makes us human, and AI is an extension of that quality.”

— Yann LeCun, Professor at New York University

Abstract

Xylella fastidiosa (Xf) is a plant pest able to infect over 500 plant species worldwide. This pathogen has already caused considerable economic and environmental damage to olive groves in Apulia (Italy) in recent years, and has since spread throughout Mediterranean coastal zones. However, there are no effective control strategies against it and the European Commission currently establishes hard eradication measures in the some of the most affected regions. Particularly, all susceptible plants that are within a radius of 100 meters around an infected specimen must be uprooted, resulting in a great economic loss. Consequently, diverse techniques and methods have been developed to detect the presence of *Xylella fastidiosa* in crops and monitor its spatio-temporal spreading dynamics in a large scale in order to prevent its expansion and impact. Traditional infield survey methods are accurate but costly for regional studies and monitoring. Instead, remote sensing along with machine learning algorithms constitute a quick and cost-effective methodology for determining the presence of the disease. Hence, in this project we present a novel technique for automatic detection of *Xylella fastidiosa* from satellite imagery. Particularly, we employ WorldView-2 satellite imagery with their 8-band multispectral data and a selection of vegetation indices for the purpose of training selected machine learning algorithms (SVM, artificial neural networks, recurrent neural networks, etc.) to determine whether an almond tree has the disease or not. The pilot testing has been carried out in Son Cotoner d'Avall farm (Puigpunyent, Mallorca), where a sample of 749 almond trees have been subjected to q-PCR tests for *Xylella fastidiosa* during 2018, wherefrom we are provided with a WorldView-2 satellite image dated 22 June 2011. The applied multidisciplinary approach is promising, as the trained algorithms show accuracies above 65% despite of the time lag between the *Xylella fastidiosa* tests and the satellite image. Therefore, this work shows that large-scale satellite Xf monitoring is feasible and opens the possibility of significant and promising progress based on this idea.

Preface

This work carried out by the master student Javier Galván Fraile constitutes his Master's Thesis at the Master's Degree in Physics of Complex Systems of the Universitat de les Illes Balears. It has mainly been developed at the Institute for Cross-Disciplinary Physics and Complex Systems (IFISC) under the mentoring of Dr. José Javier Ramasco Sukia and Dr. Manuel Alberto Matías Muriel.

This project constitutes the basis for future work on the field of *Xylella fastidiosa* detection using remote sensing. Particularly, it presents the problem about the propagation of the *Xylella fastidiosa* spreading and its social, ecological and economic damage. Once the real concern about this disease is presented, a review of the state of the art about the methods employed in its detection and prevention is presented along with the advantages and disadvantages of each. Then, the idea of satellite remote sensing is presented and tested on Son Cotoner d'Avall farm (Puigpunyent, Mallorca) where a sample of 749 almonds have been subjected to q-PCR tests for *Xylella fastidiosa* during 2018. Moreover, a WorldView-2 satellite image dated 22 June 2011 is used to obtain the high-resolution panchromatic as well as 8-band multispectral reflectance together with a selection of vegetation indices. Consecutively, a digitisation of the almond trees via photointerpretation was done, assigning each pixel to its corresponding tree and labelling it as infected, non-infected or agricultural land. Finally, different machine learning algorithms (SVM, artificial neural networks, recurrent neural networks, etc.) were trained and tested over this sample of almond trees. Therefore, the aim of this project is to settle the basis for the application of machine learning techniques to satellite remote sensing of *Xylella fastidiosa*.

Acknowledgements

I would firstly like to thank Staff Researchers Manuel Matías Muriel and José Ramasco, because without their direction and support I would not have been able to carry out this Master Thesis. Their great knowledge in biological modelling and epidemic spreading, along with their ability to transmit it and all their support, has allowed me to discover these exciting disciplines throughout these months of intense work in both the scientific and personal fields. I also want to thank Associate Professor Joan Bauzà for all his work and guidance in everything related to remote sensing where his recognized experience has been invaluable. Besides, I want to thank the Conselleria d'Agricultura del Govern Balear for providing us the results from the q-PCR study performed on the almond trees from the Son Cotoner d'Avall farm, as well as Eduardo Moralejo for his advices on the biological perspective of the *Xylella fastidiosa* epidemic. Lastly, I want to make a special mention to Dr. Andrew Ng (Standford University) for providing me the tools required to learn Machine Learning from scratch.

To my fellow students, friends and all the people who have contributed to my personal growth in this unique year, thank you.

To my dear friends and flatmates, comrades of happiness, quarantine and fatigue, Ana, Jorge and Medi, thank you and best wishes for the time ahead..

Finally, to my mother, father and brother because they have always been there when I needed them, thank you.

List of abbreviations and terms

Xf	*Xylella fastidiosa*
WV2	WorldView-2
qPCR	Quantitative Polymerase Chain Reaction
NIR	Near-infrared
TOA	Top-of-atmosphere
UTM	Universal Transverse Mercator
PCA	Principal Component Analysis
SVM	Support Vector Machine
ANN	Artificial Neural Network
RNN	Recurrent Neural Network
LSTM	Long Short-Term Memory

Table of Contents

List of Figures

List of Tables

1. Global overview and state of art

Abstract

In this chapter, a brief introduction to *Xylella fastidiosa*, the bacteria that threatens global agriculture, is presented. For that purpose, a review of the field bibliography is made with particular emphasis on the agricultural, environmental, cultural and economic consequences of the *Xylella fastidiosa* outbreak in the region of Apulia (Italy). Hereafter, a quick review of the main efforts to overcome *Xylella fastidiosa* plague is presented just to place the reader in the state of the art scenario. Finally, our proposal is presented as an emergent solution to the drawbacks of the different countermeasures previously outlined.

1.1 *Xylella fastidiosa*: A real concern

Xylella fastidiosa, a xylem-limiting bacterium, is a plant pathogen that affects a large number of species, more than 500, including high value crops, on which it causes large economic losses [Pob+20]. This bacterium has three subspecies with presence in the Balearic islands (*multiplex*, *pauca*, *fastidiosa*). Affected species by the bacterium include grapevines (Pierce's disease), citrus variegated chlorosis, coffee, almond and olive trees. Originally it was distributed in the American continent [Aut+15], but more recently it has spread to other continents, having been detected in places like Iran and Taiwan. In Europe it was first detected in Southern Italy (Apulia), in October 2013, where it has caused large damage in olive groves [Alm16], and later in Southern France and Spain (both in the Balearic Islands and the mainland), attacking almond trees (the Almond Leaf Scorch Disease, ALSD). In the case of the Balearic Islands the first report is from the fall of 2016, although it is suspected that the bacterium was introduced much earlier, even in the last years of the previous century [MP19]. In this respect, recent studies suggest that at least one million almond trees in Mallorca have been infected by the plague in the last two decades, which constitutes nearly 70% of this crop on the island [Mal20] .

The spread of X. fastidiosa has not been contained in Europe and there is concern that it might devastate, e.g., the very big olive tree plantations in Andalucia, as it has done already in the Italian Apulia. Transmission between one affected tree to a healthy one occurs

Figure 1. Evolution of X. fastidiosa in an olive grove in the Apulia region (Italy) [Alm18].

through some insect species, including the meadow spittlebug (Philaenus spumarius), that have a stylet they use to feed on xylem sap (see Figure 2). In this process they can pass X. fastidiosa from a tree with the disease to a healthy one. There is no treatment against the disease at the moment, and cutting down affected trees seems the only strategy. This is made more difficult by the fact that a tree may show little or no symptoms, and still being infectious for more than 5 months [Alm16]. Throughout this incubation period, the disease causes water-related stress leading to lower transpiration and photosynthetic rates. Once the symptoms become visible the tree tops start to present discolouration and defoliation as stages previous to the tree's death [Hor+20]. The most sensitive technique to detect the diseases originated by the X. fastidiosa is the q-PCR (quantitative Polimerase Chain Reaction) test. However, the effectiveness of the technique in field conditions is limited by the sampling period and the uneven distribution of the bacterium in a canopy of affected trees, particularly in the asymptomatic stage. In addition, this laboratory technique is costly and time-consuming, and requires skilled and trained personnel [Zar+18].

On account of the destruction it leaves on its wake and the absence of an effective treatment, *Xylella fastidiosa* is considered a regulated quarantine pest. In particular, it was estimated that the annual cost of *Xylella fastidiosa* for the wine industry in California amounted to $104.4 million per year in 2014 if we add up the costs of vine losses, industry assessments, compliance costs, and expenditures by government entities [T+14]. Likewise, the stay of the disease in Italy has caused an agricultural, environmental and economic disaster. According to the European Union, it has razed a million olive trees and put 300,000 jobs at risk [Vid20]. As a response, the European Commission approved in February 2014 countermeasures to prevent the introduction and spread of X. fastidiosa. These measures include elimination of infected trees and other susceptible/asymptomatic hosts within 100 meters, the use of treatments to suppress vector populations, and monitoring of areas surrounding the locations where the disease has been found. However, these measures

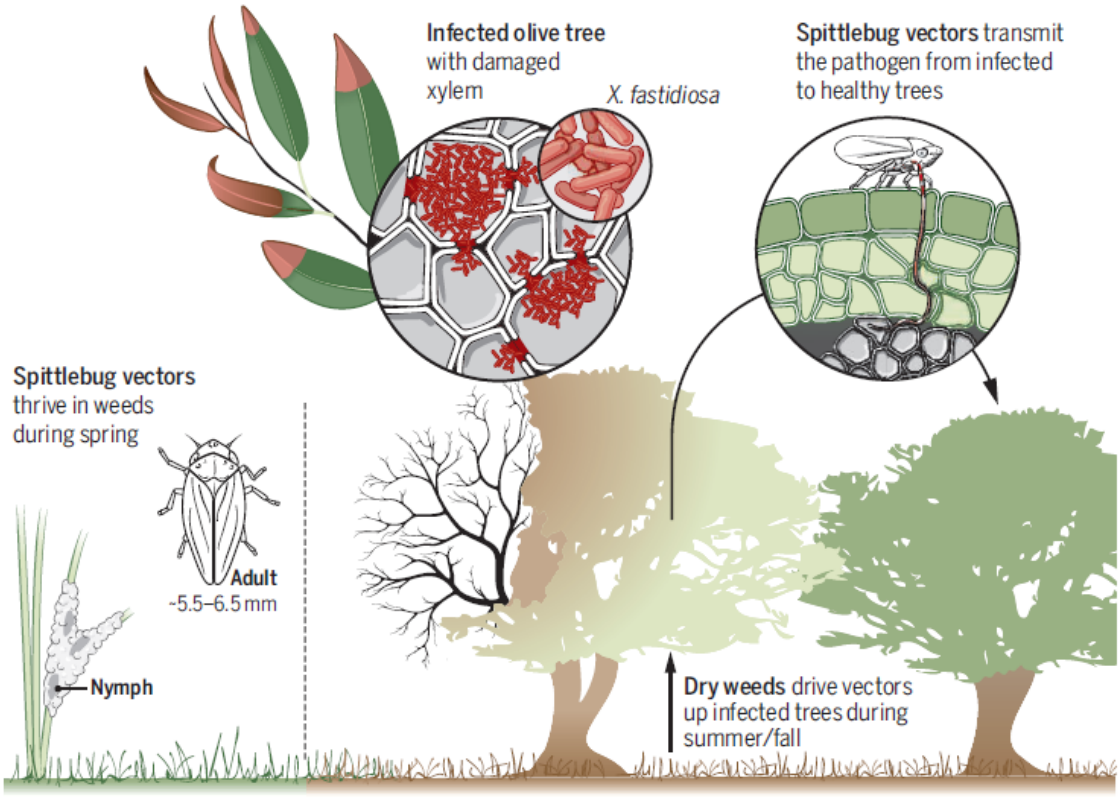


Figure 2. Epidemiology of *Xylella fastidiosa* infection. The prevailing hypothesis suggest that the trees are the main pathogen inoculum source, and that the spreading of the disease is carried by insect vectors [Alm16].

have not been followed by all the European countries [Alm16].

1.2 Related work

The current eradication and containment measures for X. fastidiosa are based in the early detection of infected plants through a large scale monitoring, which mostly relies on visual field surveys and subsequent laboratory analyses [Alm16]. Particularly, apart from the visual inspection of the crops done by specialized technicians, in the last year a total of 7,339 q-PCR have been realized [eur20]. In general, the ground methods used in the detection of plant diseases in the crops (like q-PCR test) are prohibitively expensive. Henceforth, new alternatives which enable a large scale monitoring are required.

In this regard, the use of remote sensing constitutes an interesting way of performing robust monitoring and early disease detection in plants. Most of the research that follow this approach are based on information obtained with manned and unmanned flights, equipped with multiespectral and hyperespectral sensors, as well as thermal cameras [Pob+20][Zar+18]. This technique is relatively expensive because of the need of regular flights and, moreover, it is only able to provide the relevant information from the moment

in which the flights are carried out, unlike to what happens with satellite imagery.

In the present proposal we are suggesting a novel procedure to surpass these drawbacks with the use of satellite imagery, which constitutes a cheap way of monitoring the outbreak of X. fastidiosa on some species (almond-trees in our study). This approach is widely employed in high precision agriculture, allowing, for example, the monitoring of farms in detail, as well as providing more intensive and efficient cultivation practices. Also, satellite imagery have already been used to detect some outbreaks in agriculture, like yellow rust in wheat. The reported research study the differential absorption in selected spectral bands and synthetic indices when the crop is healthy versus when it has been affected by the disease [Lia+18]. Hence, these studies lay the foundations for the present project.

This approach comes along with the use of Machine Learning techniques in order to build up the relevant combination of spectral bands and standard indices able to predict the degree of affectation induced by the disease. This idea is based on the fact that the hydric stress induced by the clots produced by the aggregation of bacteria in biofilms inside the xylem vessels changes the absorption pattern in spectral bands, affecting the chlorophyll absorption. Thus, a deep learning approach is able to use all the available bands to improve the predictability. The long-term goal is, of course, being able to characterize the spectral bands or synthetic indices that are more useful in predicting the disease in various trees.

1.3 Why Machine Learning?

Machine learning techniques have already proven to be really efficient in several tasks related to the multi-disciplinary agritechnologies domain. Particularly, these techniques have been used in crop management, livestock management, water management, yield prediction, disease detection, among others [Lia+18]. Also, at it was abovementioned, infield methods turn out to be really expensive and it can be difficult to reach some trees by land when we are considering not accessible regions, like ravines and gorges. Furthermore, it takes a long time to analyze all the trees individually; the larger the number of trees, the longer the time and cost required. These subtleties can be overtaken if satellite imagery is used instead of these more traditional and manual approaches. With this in mind, it must be pointed out that some investigations have already used the power of machine learning applied to satellite imagery in the framework of plant diseases with promising results [PEL17].

2. Introduction to Machine Learning

Abstract

This chapter constitutes a condensed review of what Machine Learning is, with special attention in the supervised learning subfield. Concretely, several machine learning models are presented covering from traditional algorithms like Support Vector Machine (SVM) to fresher and deeper structures like Artificial Neural Networks (ANN) and Recurrent Neural Networks (RNN). Finally, a selection of the most interesting classification metrics for binary classification problems is presented.

2.1 Machine Learning models

Machine learning, defined as the field of study that gives computers the ability to learn without being explicitly programmed, is classified as a subfield of Artificial Intelligence [Sam59]. In general, a supervised machine learning problem can be formulated as follows. Given a training dataset $\{(\mathbf{x}_1, y_1) \dots (\mathbf{x}_m, y_m)\}$ in $\mathbf{R}^{\mathrm{n}} \times \mathbf{R}$ sampled according to an unknown probability distribution $P(\mathbf{x}, y)$, and a cost function $\mathcal{L}(y, f(\mathbf{x}))$, for a given $\mathbf{x}$ a value $\hat{y} = f(\mathbf{x})$ is predicted instead of the truth value y. Then, the supervised machine learning problem consists in finding a function f, also known as *hypothesis*, that minimizes the expectation of the cost on data never seen by the algorithm. This is known as *generalization*.

According to the literature, many different machine learning algorithms have been used in agriculture (artificial neural networks, bayesian models, dimensionality reduction, support vector machines, etc.). In this project we will employ traditionally used supervised learning algorithms (like SVM) and emerging deep learning architectures (like deep neural networks) as we expect them to be the ideal mechanism to extract the complex combination of bands and indices that might characterize better the presence of X. fastidiosa.

2.1.1 Support Vector Machine (SVM)

Support Vector Machine (SVM) is a classifier that uses machine learning background to maximize predictive performance while avoiding overfit to the data [Ope20]. Specifically, it

tries to find the hyperplane that gives the largest minimum distance to the training examples. Twice this distance is known as *margin* and, consequently SVM tries to maximize it (see Figure 3) [Jak06].

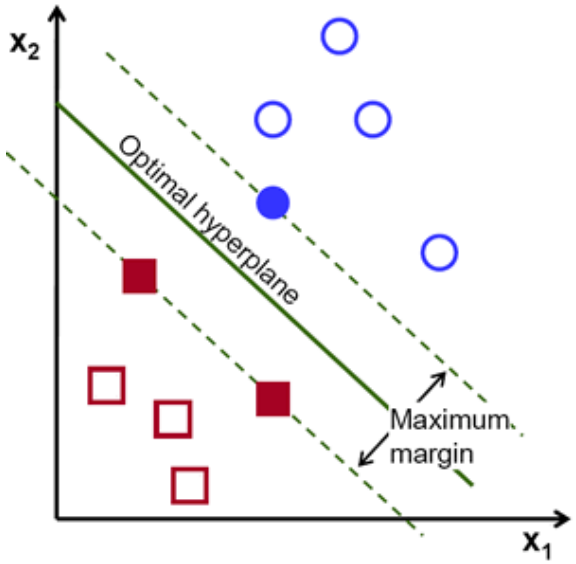


Figure 3. Optimal hyperplane in SVM classifier. Notice that it maximizes the margin between the two classes set of points [Ope20].

Firstly, we rewrite the training data as $X \in \mathcal{R}^{n \times m}$. Now, a hyperplane (θ, θ_0) able to perform a linear separation of the training data will satisfy

$$\theta^T \mathbf{X} + \theta_0 = 0, \tag{2.1}$$

where $\theta \in \mathcal{R}^n$ is the weight vector and $\theta_0 \in \mathcal{R}$ the bias. For the case of a linearly separable training set, an *optimal hyperplane* is the one that separates the positive and negative samples without error while maximizing the distance with the closest training samples. Now, we define the *canonical hyperplane* as an optimal hyperplane satisfying

$$\begin{aligned} &\text{If } y_i = 1: \quad \theta^T \mathbf{x_i} + \theta_0 \geq 1 \\ &\text{If } y_i = 0: \quad \theta^T \mathbf{x_i} + \theta_0 \leq -1. \end{aligned} \tag{2.2}$$

This constraint can be rewritten as

$$|\theta^T \mathbf{x_i} + \theta_0| \geq 1; \qquad i = 1, \ldots, m \tag{2.3}$$

where the equality is satisfied for the closest training examples to it. Now, recall that the distance, d, between a given point $\mathbf{x_i}$ and a hyperplane (θ, θ_0) corresponds to

$$d = \frac{|\theta_0 + \theta^T \mathbf{x_i}|}{||\theta||}, \tag{2.4}$$

where $||\ldots||$ is the vector norm. Particularizing this expression for the canonical hyperplane and the training examples $\mathbf{x_c}$, we get the distance d_c given by

$$d_c = \frac{|\theta_0 + \theta^T \mathbf{x_c}|}{||\theta||} = \frac{1}{||\theta||}. \tag{2.5}$$

Notice that twice this distance is the margin, $\mathcal{M}$:

$$\mathcal{M} = 2d_c = \frac{2}{||\theta||}. \tag{2.6}$$

Finally, we are aiming to maximize the margin, which is equivalent to minimize the following cost function $J(\theta)$ given by

$$\min_{\theta,\theta_0} J(\theta) = \min_{\theta,\theta_0} \frac{1}{2}||\theta||^2, \tag{2.7}$$

subjected to the constraints (2.2). These constraints may look arbitrary, but they are introducing the large margin intuition[1]. Then, the problem reduces to a Lagrangian optimization which can be solved employing Lagrange multipliers, among others constrained optimization methods. However, in many cases we cannot perform a perfect linear separation and we must look for the hyperplane that maximizes the margin while minimizing the misclassifications. For that purpose, we introduce the variable ζ which allows some objects to fall of the margin with a penalization (see Figure 4).

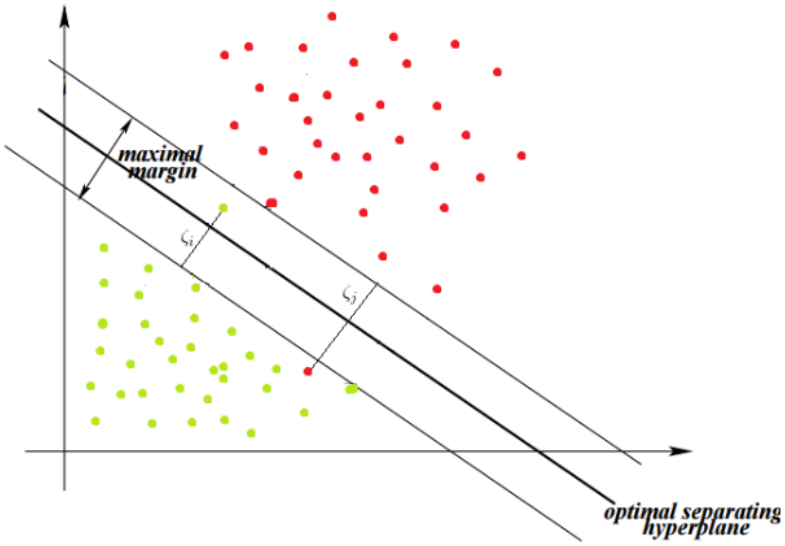


Figure 4. Effect of the regularization parameter in the SVM algorithm [Jai17].

[1]For further details and discussion about this particular choice check the following post: https://stats.stackexchange.com/questions/193444/what-is-the-purpose-of-1-1-constraint-on-svms.

Now, our optimization problem can be rewritten as

$$\underset{\theta,\theta_0}{min} J(\theta) = \underset{\theta,\theta_0}{min} \frac{1}{2}||\theta||^2 + C\sum_{i=1}^{n} \zeta_i, \tag{2.8}$$

with the constraints

$$\begin{aligned} &\text{If } y_i = 1: \quad \theta^T \mathbf{x_i} + \theta_0 \geq 1 - \zeta_i \\ &\text{If } y_i = 0: \quad \theta^T \mathbf{x_i} + \theta_0 \leq -1 + \zeta_i. \end{aligned} \tag{2.9}$$

We denote parameter C as the **regularization parameter**, and it accounts for the trade-off between the width of the margin and the number of misclassifications.

Kernels

The explained SVM method works well when the data is laid out in a linear way and thus a hyperplane can be used. However, generally the data presents a nonlinear arrangement. In order to deal with this situation, kernels are introduced as a non-linear mapping of the data to a high-dimensional space which may be linearly separable (see Figure 5).

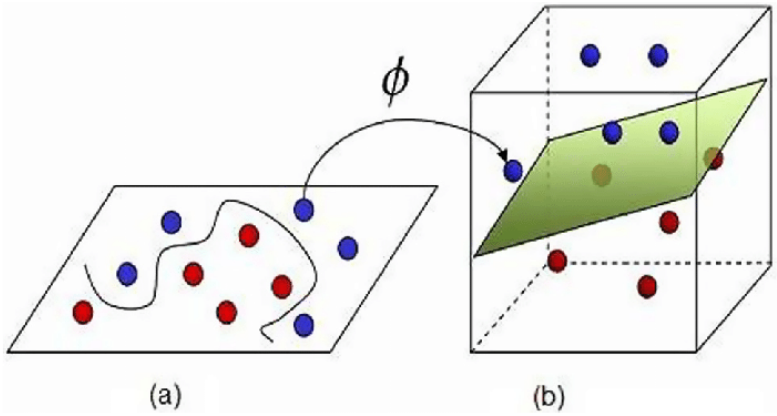


Figure 5. Kernel idea of mapping to a high-dimensional space to perform a hyperplane separation [NP16].

The idea is then to substitute vector $\mathbf{x}$ for a vector of similarities $\mathbf{f}$ defined as follows

$$f_i = K(x, x^{(i)}), \tag{2.10}$$

where K represents the kernel and $x^{(i)}$ the i^{th} training sample. Then, the equation (2.1) can be rewritten as

$$\theta_0 + \theta^T \mathbf{f} = 0, \tag{2.11}$$

and the optimization problem is the same as in (2.8) and (2.9) but using $\mathbf{f}$ instead of $\mathbf{x}$.

Our particular choice for the kernel would be the Gaussian kernel, defined as

$$f_i = K(x, x^{(i)}) = exp\left(-\frac{||x - x^{(i)}||^2}{2\sigma^2}\right), \tag{2.12}$$

because it has given superior results in the *Xylella fastidiosa* remote sensing problem, proving to be one of the best machine learning algorithms in this particular task [Pob+20].

2.1.2 Artificial Neural Networks (ANN)

The origin of Artificial Neural Networks (ANN) is associated with the idea of designing an algorithm that tries to mimic the nervous system of biological organisms[2] (see Figure 6). Mainly, they are composed of computation units connected among themselves through weights, similarly to what happens with neurons and synaptic connections. In this sense, ANN can be seen as computational graphs of basic computational units in which greater predictive power is achieved by connecting them in certain ways. With this architecture, ANN compute a function of the inputs by propagating the calculated values from the input neurons to the output neurons, and the process of learning consists in varying the weights in order to minimize a cost function of the output values and the inputs truth labels [Agg18]. With regard to this scheme, ANN have the potential of approximating any continuous nonlinear functions.

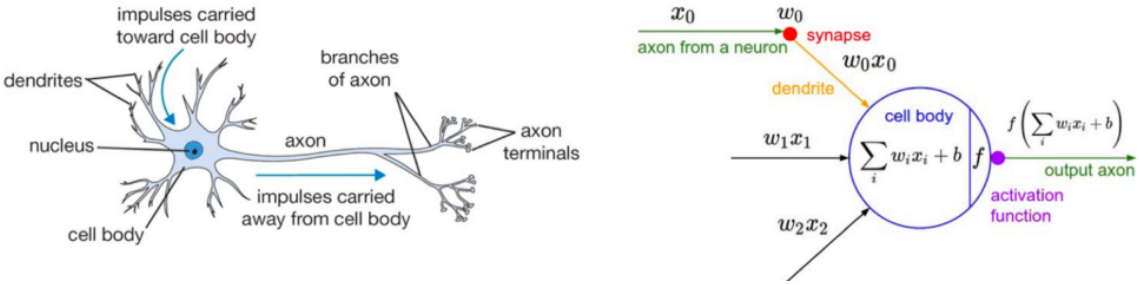


Figure 6. Artificial Neural Network vs Human Brain Processing [Dik19].

Basic computation unit

Consider a given neuron (or node) which has the input values $\mathbf{x} \in \mathcal{R}^n$. Then, the scalar product of the input values and the weights vector is determined, $\mathbf{w} \in \mathcal{R}^n$, and this quantity is added to a bias value, $b \in \mathcal{R}$, and finally an activation function, f, is applied. Hence, the output value (or activation), $a \in \mathcal{R}$, is given by

$$\mathbf{a} = f\left(\mathbf{w}\mathbf{x} + b\right). \tag{2.13}$$

[2]This biological comparison is usually criticized because of the oversimplified vision of the brain functioning it gives. Although, neuroscience investigation has provided fresh and useful ideas in designing new neural network architectures.

Activation functions

One of the key points of ANN's power resides in the non-linearity of the activation functions. Particularly, we will employ two types of activation functions:

- **Sigmoid**. The sigmoid activation function is useful for binary classification as it outputs a value in $[0, 1]$, and is given by

$$f(z) = \frac{1}{1 + e^{-z}}. \tag{2.14}$$

- **ReLU**. The ReLU activation function has replaced the sigmoid activation function in deep neural networks because of the computational speed gain in training these architectures with this replacement, and its expression is

$$f(z) = \max(0, z). \tag{2.15}$$

Multi-layer neural network

A multi-layer neural network consists in connecting a set of computation units with a given layout. In Figure 7 we can observe an illustrative example of a 2 layer ANN where the forward propagation of the input values is shown. More hidden layers can be added to the architecture resulting in even more complex features. However the forward propagation of the input values will follow the same scheme. Notice that to obtain a binary classification output ($\{0, 1\}$) it is convenient to place a sigmoid activation function in the output layer of the ANN.

Loss function and backward propagation

Up to this point we know how to obtain the binary classification prediction from our ANN. Now, it is time to evaluate the prediction made. For that purpose, we define a cost function over the training set and update the weights of the different layers by minimizing it. This process of update, based on the derivatives of the cost function, is widely known as *backpropagation*. Many different cost functions can be chosen depending on the nature of the problem and the kind of results we are aiming for. Moreover, the most commonly used cost function in binary classification problems is the **binary cross-entropy**. This classification metric provides fine details on the classifier performance and we will use it as cost function of every neural network we train. Given the output of the model, $\hat{y}_i \in [0, 1]$, over sample i and the truth label, $y_i \in \{0, 1\}$, for that sample, the binary cross-entropy, $\mathcal{L}$,

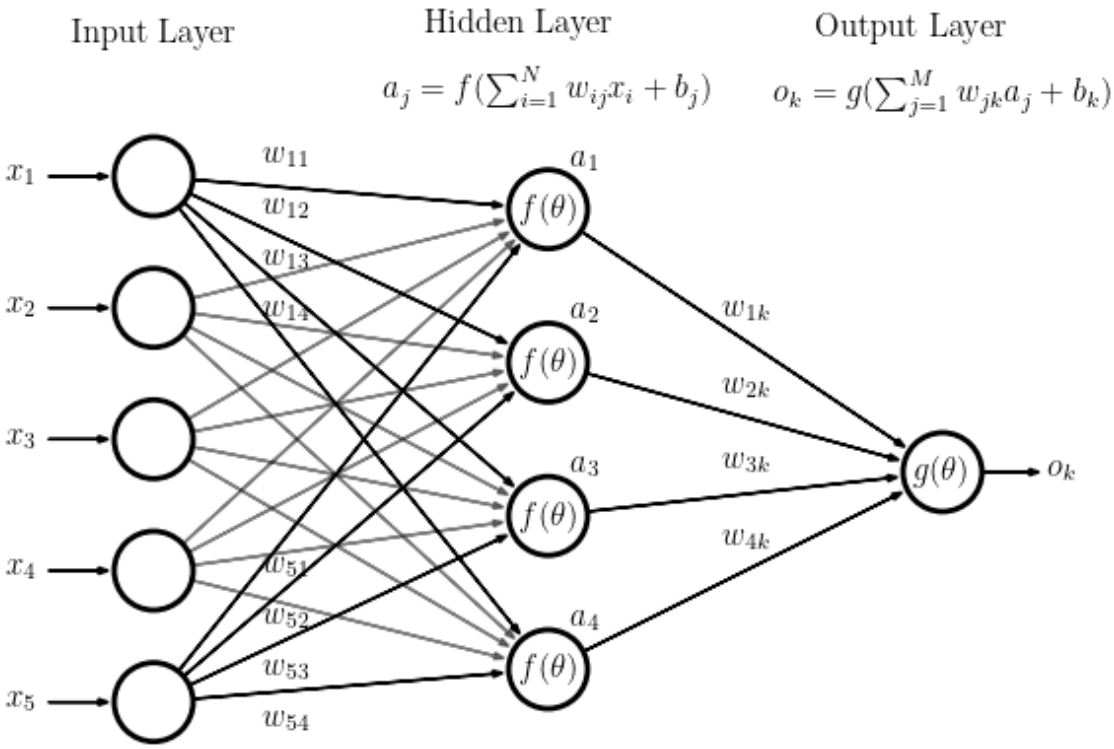


Figure 7. 1-layer Artificial Neural Network diagram [Dik19].

would be given by

$$\mathcal{L}(y, \hat{y}) = -\frac{1}{m}\sum_{i=1}^{m} y_i \log \hat{y}_i + (1 - y_i) \log (1 - \hat{y}_i)\,, \tag{2.16}$$

where m represents the size of the considered dataset. Remember that $\hat{y}_i$ represents the probability of sample i being positive. In the case of the ANN, $\hat{y}_i$ corresponds to the output value of the output layer. Note that the binary cross-entropy has no upper bound and exists on the range $[0, \infty]$, where values close to 0 mean a high accuracy.

Regularization

Throughout the training process our neural network will get better at predicting over the training set and will start to make worse predictions over the test set. This phenomena is known as *overfitting* and there are several ways to prevent it.

- **L2 Regularization**. This method reduces the magnitude of the neural network weights in order to get a simpler hypothesis which may be less prone to overfit the training set. To achieve this, a new term is introduced into the cost function (2.16)

as follows

$$\mathcal{L}(y, \hat{y}) = -\frac{1}{m}\sum_{i=1}^{m} y_i \log \hat{y}_i + (1 - y_i) \log (1 - \hat{y}_i) + \frac{\lambda}{2m}\sum_{l=1}^{L} ||\mathbf{w}^{[l]}||_F^2 \quad (2.17)$$

where λ is the regularization parameter, the second sum extends over all the neural network layers and $||\dots||_F$ is the Frobenius norm.

- **Dropout regularization**. The idea of the dropout technique is to eliminate with probability $p^{[l]}$ some nodes from the neural network layer l in each example during the training. This procedure causes the neural network weights to shrink. During the testing time dropout can not be used.

There are some concerns with the use of artificial neural networks (ANN) such as finding how many neurons are needed for a given task as well as the ANN architecture. Also, there may not exist a unique solution of the problem as there may be many linear classifiers (hyperplanes) which can classify accurately the data. These are the main advantages of SVM over ANN. Otherwise, an ANN will outperform a SVM when there is a large training set. However, it should be borne in mind that there is no better model over the full range of problems.

2.1.3 Recurrent Neural Networks (RNN)

Recurrent Neural Networks (RNN) are a class of neural networks that take as input both current input example and the ones that have already been seen by the use of hidden states. This kind of architecture is quite useful when dealing with sequence of data (speech recognition, sentiment classification, machine translation, etc.). In this situation, they show some advantages over ANN:

- Inputs and outputs can have different lengths in different examples.
- Unlike ANN, RNN share features learned across different input positions.

The main idea behind RNN networks is the fact that information from one input influence the predictions over future inputs prediction on the same sample. Hence, given an input sequence $\mathbf{x} = \{x^{<1>}, \dots x^{<t>}, \dots x^{<T_x>}\}$ with $x^{<t>} \in \mathcal{R}^n$, and an output sequence $\mathbf{y} = \{y^{<1>}, \dots y^{<t>}, \dots y^{<T_y>}\}$ with $y^{<t>} \in \mathcal{R}$, the general architecture of RNN is shown in Figure 8.

From Figure 8 we observe that for each time step t, the activation $a^{<t>}$ and the output

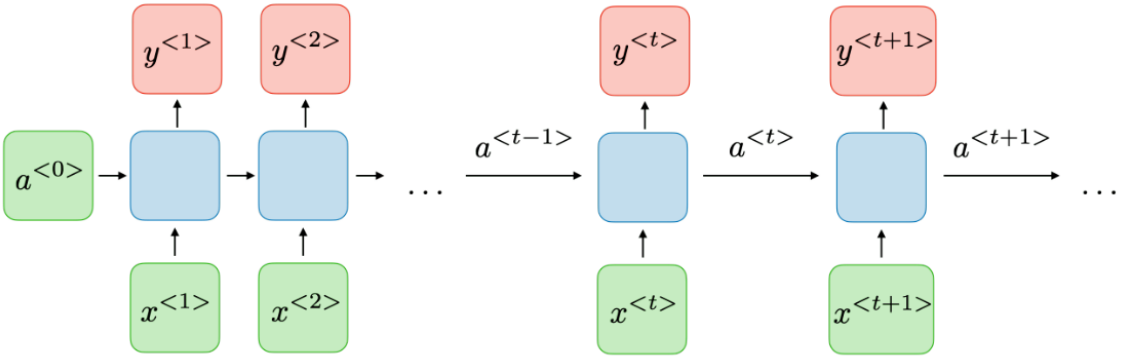


Figure 8. RNN common architecture diagram [AA20].

$y^{<t>}$ are determined considering previous activations, $a^{<t-1>}$, and the current input $x^{<t>}$. Besides, the cost function of the RNN will be given by

$$\mathcal{L}(\hat{y}, y) = \sum_{t=1}^{T_y} \mathcal{L}\left(\hat{y}^{<t>}, y^{<t>}\right), \tag{2.18}$$

which will be backpropagated in order to update the RNN weights. There exist many ways to compute these quantities of the hidden cells and here we will concern about the Long Short-Term Memory (LSTM) unit.

Long Short-Term Memory (LSTM)

LSTM unit contains information outside of the normal flow of the recurrent neural network which can be stored in, written to, read or erased. These computation units prevent the neural network from suffering the vanishing gradient problem. To achieve this, LSTM uses some specific gates with an established purpose. Particularly, gate Γ_j is given by

$$\Gamma_j = \sigma\left(W_j x^{<t>} + U_j a^{<t-1>} + b_j\right), \tag{2.19}$$

where W_j, U_j and b_j are trainable weights specific to the gate and σ represents the sigmoid activation function. In total there are 4 different gates:

- Update gate, Γ_u. It controls how much past information shall be considered.
- Relevance gate, Γ_r. It handles whether previous information should be dropped.
- Forget gate, Γ_f. It manages if a cell should be erased or not.
- Output gate, Γ_o. This gate controls how much information should be revealed of a cell.

With these ingredients we construct the LSTM unit according to the diagram shown in

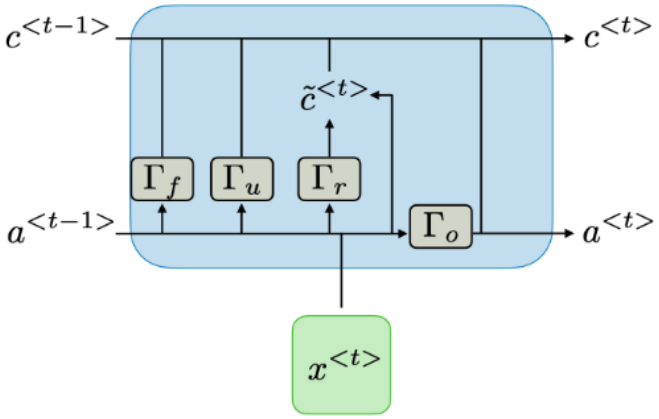


Figure 9. LSTM computation unit [AA20].

Figure 9, and the equations characterizing the different variables are the following

$$\begin{aligned} \tilde{c}^{<t>} &= \tanh\left(W_c\left[\Gamma_r a^{<t-1>}, x^{<t>}\right] + b_c\right) \\ c^{<t>} &= \Gamma_u \tilde{c}^{<t>} + \Gamma_f c^{<t-1>} \\ a^{<t>} &= \Gamma_0 c^{<t>} \\ y^{<t>} &= g_2\left(W_{ya} a^{<t>} + b_y\right) \end{aligned} \tag{2.20}$$

Many-to-one architecture. Sentiment classification

Among the different types of RNN we will focus on the Many-to-one architecture ($T_x > 1$ and $T_y = 1$). This architecture, shown in Figure 10, is commonly used in sentiment classification problems.

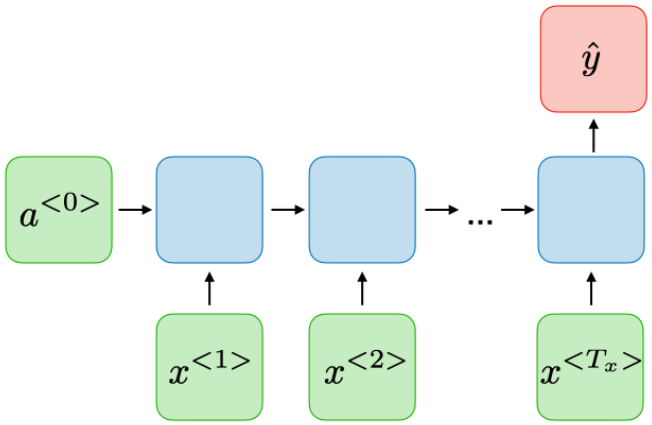


Figure 10. RNN many-to-one architecture diagram [AA20].

This kind of problems consist in identifying opinions in text and labeling them as positive,

negative and neutral according to the emotions the user shows in them. In this task, the information given by the first words of the sentence may be crucial for identifying the correct emotion:

"@wcve it's amazing how our city loves him and he really loves our city. @HillaryClinton made a great choice for Vice President. @timkaine".

In our case, as every tree will be formed by an undefined number of pixels and the disease may be present only in a fraction of them, we can use this architecture to exploit this fact and transmit the strong signs of disease found in one pixel to the rest of the network.

Bidirectional LSTM RNN

This type of architecture constitutes an extension of traditional single LSTM may help to enhance the fact that the disease can be only in some pixels by boosting the weight of the information of these pixels. The idea is the same as in the single LSTM network but considering now another LSTM unit that propagates backwards (see Figure 11).

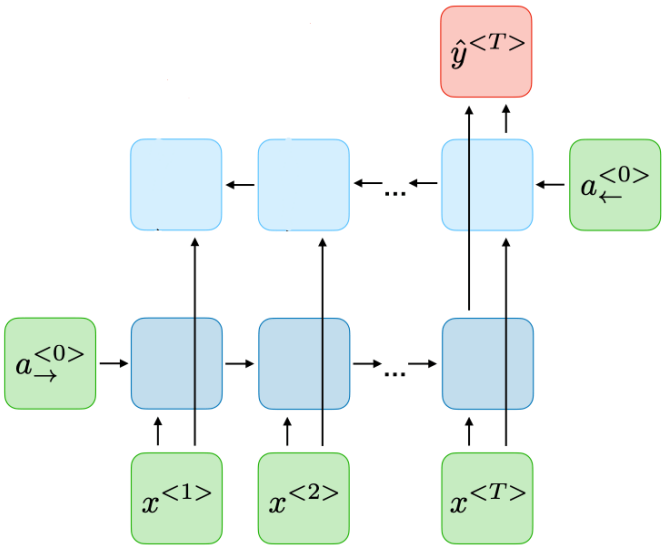


Figure 11. Bidirectional LSTM RNN many-to-one architecture diagram [AA20].

2.2 Classification metrics

Every machine learning task requires from an evaluation metric in order to quantify the performance of the model. In our case, deciding whether a plant is infected by *Xylella fastidiosa* or not constitutes a binary classification problem, i.e. the output of the model will be 0 (non-infected) or 1 (infected). Subsequently, we will present several ways of measuring the success of a model.

Accuracy

This metric consists in the ratio of number of correct predictions to the size of the validation set (total number of predictions):

$$\text{accuracy} = \frac{\text{\# correct predictions}}{\text{Total number of predictions made}}. \tag{2.21}$$

However, it works well only if there are an equal number of samples belonging to each class [Zhe15].

Confusion matrix

On account of its simplicity, accuracy penalizes in the same way the errors committed when misclassifying any of the two classes and, in some cases, the cost of failing to diagnose a positive case can be much higher than making wrong prediction (consider for example the cost of failing to diagnose the disease of a person vs the cost of sending a healthy person to take more tests) [Mis18]. Otherwise, a **confusion matrix** presents more detailed information about the model performance (see Table 1).

		Model prediction		
		Positive	Negative	Total
Truth label	Positive	TP	FN	$TP + FN$
	Negative	FP	TN	$FP + TN$
	Total	$TP + FP$	$FN + TN$	

Table 1. Confusion matrix diagram.

Precision-Recall

Precision shows the number of correct positive predictions divided by the total number of positive predictions made by the classifier. It is mainly answering the question: *Out of all positive predictions, how many are truly positive?*, and is given by

$$\text{precision} = \frac{\text{\# correct positive predictions}}{\text{Total number of positive predictions made}} = \frac{\text{TP}}{\text{TP+FP}}. \tag{2.22}$$

Otherwise, **recall** (also known as sensitivity or True Positive Rate) represents the number of correct positive predictions divided by the total number of examples labelled as positive. Hence, it answers the question: *Out of all positive samples, how many are identified?*. Consequently, its expression is

$$\text{recall} = \frac{\text{\# correct predictions}}{\text{Total number of positive examples}} = \frac{\text{TP}}{\text{TP+FN}}. \tag{2.23}$$

F1-Score

This classification metric tries to find balance between precision and recall by taking the harmonic mean between them. It merges these two metrics and tells us how precise (number of instances correctly classified) and how robust (commit a low number of misclassification) our algorithm is performing [Zhe15]:

$$\text{F1-Score} = \frac{2}{\frac{1}{\text{precision}} + \frac{1}{\text{recall}}}. \tag{2.24}$$

ROC and AUC

Before introducing these two concepts we firstly define the **False Positive Rate** as the proportion of negative samples that are labelled as positive with respect to the total number of negative samples:

$$\text{False Positive Rate} = \frac{\text{FP}}{\text{FP+TN}}. \tag{2.25}$$

Then, the receiver **operating characteristic curve (ROC curve)** shows how many correct positive predictions (TP) can be gained as we vary the classification threshold (commonly set to 0.5) allowing more false negative predictions (FN). The best posible performance require hitting a 100% rate inmediately without committing any false negatives. The **Area Under Curve (AUC)** appears as a single number metric that encapsulates the ROC information. Particularly, it represents the probability that a classifier will rank a randomly chosen positive observation higher than a randomly chosen negative observation [Zhe15] [Mis18] (see Figure 12).

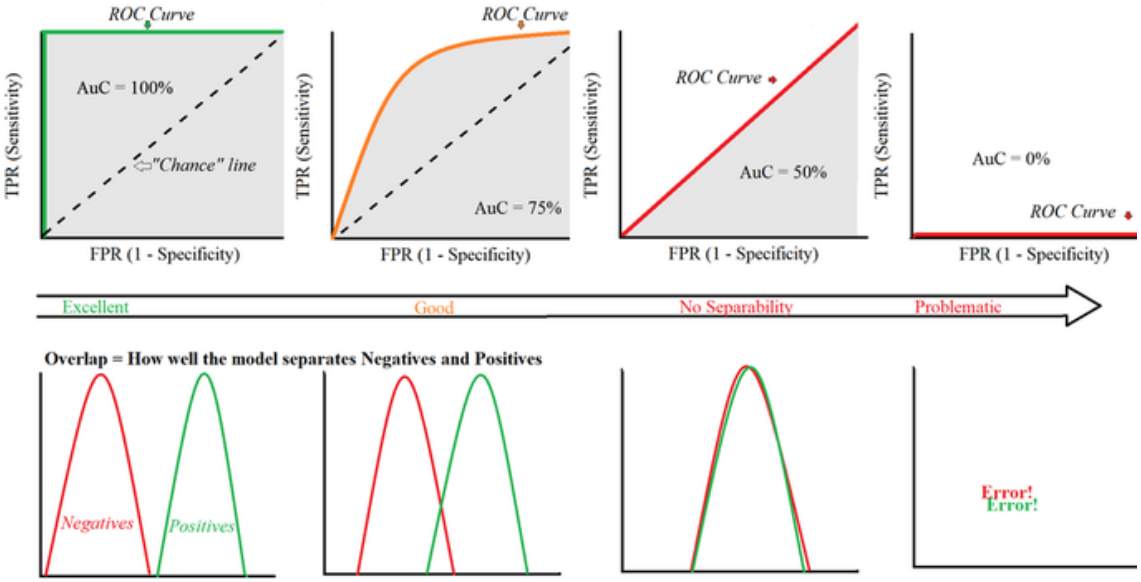


Figure 12. ROC and AUC scheme representing the variation of these concepts with the model performance. The "Chance" line represents the ROC of a random classifier [Gle19].

3. Methodologies and Applications

Abstract

The present chapter presents the main ideas and guidelines of the present project. Firstly, both the q-PCR tests and the satellite imagery of the Son Cotoner d'Avall farm (Puigpunyent, Palma) are presented. Then, the satellite imagery preprocessing is broken down into single steps: from the corrections to the satellite imagery to the vegetation indices calculation. Afterwards, the statistical properties of the 26 model features (8 bands + 18 indices) are analyzed with a Principal Component Analysis (PCA). Finally, all the Machine Learning models trained are presented in detail.

A conceptual diagram of the methodology and procedure applied in the project is presented in the Workflow diagram 23.

3.1 Material

3.1.1 Study area and field data collection

The study area was the Son Cotoner d'Avall farm, in the municipality of Puigpunyent. In this farm a sample of 749 almonds were subjected to q-PCR tests for *Xylella fastidiosa* during 2018, with the result of 272 positives and 477 negatives. These results along with coordinate information for each almond were provided by the Conselleria d'Agricultura in Microsoft Excel format. We then imported this data into a Geographic Information System and represented it on a 2015 orthophoto of the National Geographic Institute (see Figure 13).

3.1.2 Satellite imagery information

Remote sensing techniques can be useful in identifying, on a regional scale and at a low cost, potentially infected trees. The presence of *Xylella fastidiosa* in a tree creates anomalies in its capacity of chlorophyll absorption, giving rise to hydrological stress, which is usually related to changes in the infrared spectrum bands [Per+13].

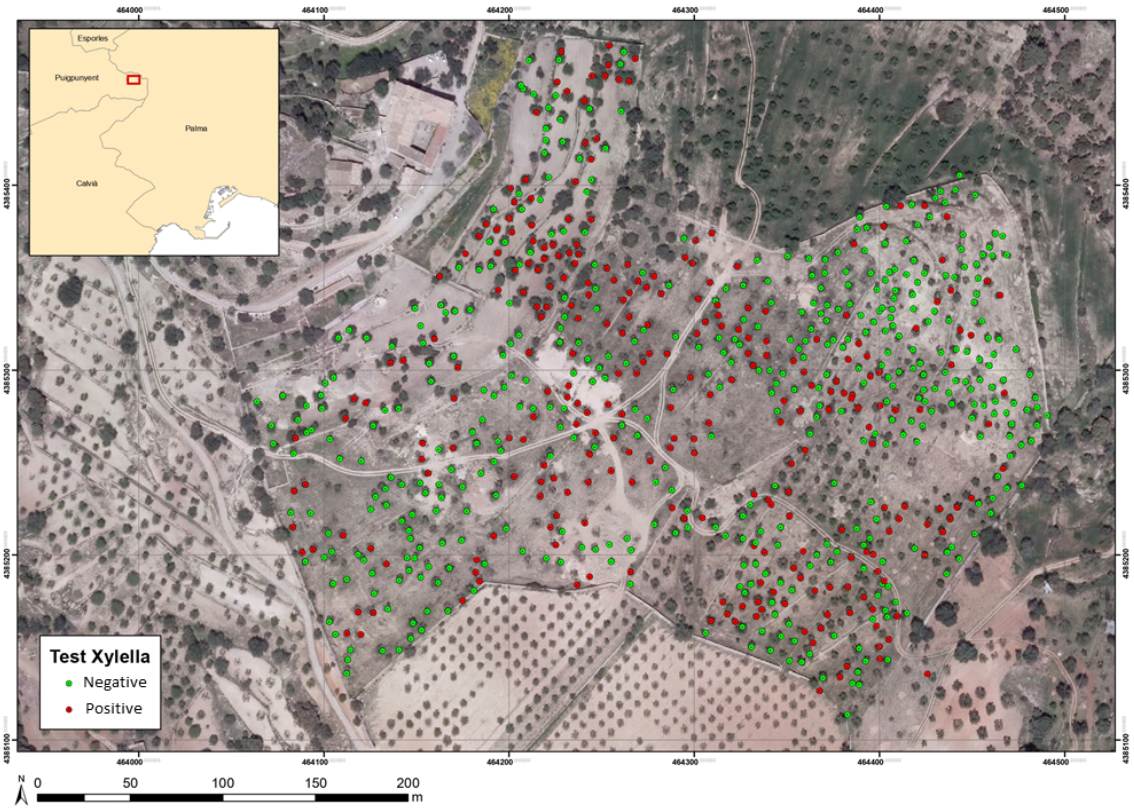


Figure 13. Distribution of *Xylella fastidiosa* test results according to dataset supplied by the Conselleria d'Agricultura. Cartographic basis: orthophoto 2015 National Geographic Institute. Projection UTM, UTM zone 31N, Datum ETRS89

To carry out the pilot test, an image from the WorldView-2 satellite dated June 22, 2011 was used. WorldView-2 is a high-resolution 8-band multispectral commercial Earth observation satellite that belongs to DigitalGlobe. It presents an orbital period of approximately 100 minutes, thus taking a new photograph of a given location every 1.1 days. Its satellite sensor has a total of eight multispectral bands with a spatial resolution (pixel size) of 1.8 m and a panchromatic band with a spatial resolution of 0.46 m [Dig10]. The relation of multispectral bands can be seen in both Figure 14 and Table 2. Notice that bands B1 to B6 correspond to the visible part of the electromagnetic spectrum and bands B7 to B8 record energy in the infrared wavelength[LL01].

3.1.3 Image preprocessing

Regarding to the WV2 image for the pilot test, it was delivered in product level LV2A, which means that it had been corrected for radiometric[1], geometric, sensor and terrain distortions. Particularly, WV2 imagery is delivered as a sample of radiometrically corrected

[1]In the visible and near-infrared wavelengths the radiance measured by WorldView-2 is dominated by reflected solar radiation[UC10].

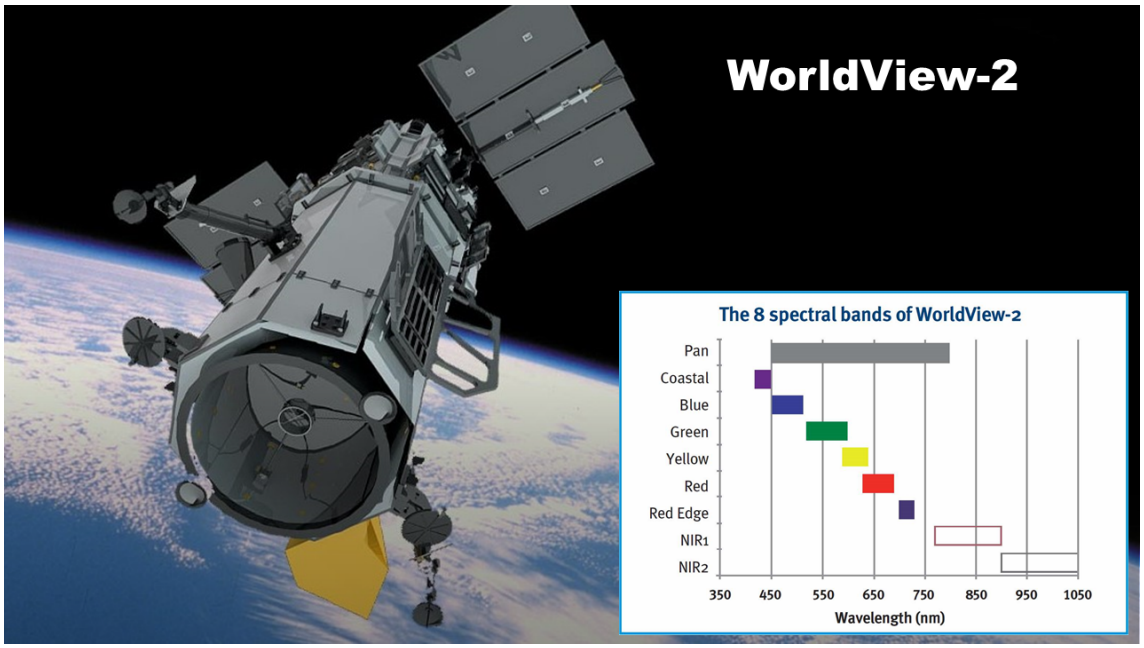


Figure 14. WorldView-2 commercial Earth observation satellite and visual comparison of its spectral bands [ESA20] [Cor20].

image pixels[2], $q_{pixel,Band}$, which are a function of how much spectral radiance enters the sensor and the instrument conversion to digital data after the abovementioned corrections. This signal depends strongly on the telescope and detector's characteristics, so image pixels should be converted to top-of-atmosphere spectral radiance in order to be compared with imagery of other sensors or with WV2 imagery collected in different conditions. This step enables the calculation of vegetation indices and transforms the data into a common scale. Top-of-atmosphere spectral radiance (TOA), $\mathrm{L}_{\lambda\mathrm{Pixel,Band}}$, is defined as the spectral radiance entering the telescope aperture at the WorldView-2 altitude of 770 km, and can be determined from the radiometrically corrected image pixels as

$$\mathrm{L}_{\lambda_{\mathrm{Pixel,Band}}} = \frac{\mathrm{K}_{\mathrm{Band}} \cdot \mathrm{q}_{\mathrm{Pixel,Band}}}{\Delta\lambda_{\mathrm{Band}}}, \tag{3.1}$$

where $\mathrm{L}_{\lambda\mathrm{Pixel,Band}}$ are top-of-atmosphere spectral radiance image pixels [$Wm^{-2}sr^{-1}\mu m^{-1}$], $\mathrm{K}_{\mathrm{Band}}$ is the absolute radiometric calibration factor [$Wm^{-2}sr^{-1}\mu m^{-1}\mathrm{counts}^{-1}$] for a given band, $\mathrm{q}_{\mathrm{Pixel,Band}}$ are radiometrically corrected image pixels [counts], and $\Delta\lambda_{\mathrm{Band}}$ is the effective bandwidth [μm] for a given band. This information is contained in the image metadata (.IMD file extension) accompanying the WV2 image [UC10]. From now on

[2]Raw detector data (DN) measured by the telescope are 11-bit data in each of the nine spectral bands and take a value out of 2048 (11-bits) possible digital numbers, which are then stored as 16 bit integers. However, a reduction of the maximum DN collected is done by the imaging companies to account for extremely reflective surfaces which could create flares. Consequently, DN values rarely exceed 1500 in raw very high resolution satellite imagery if any radiometric correction or contrast enhancement has been carried out [Agu+13].

Band	Lower Band Edge (nm)	Center Wavelength (nm)	Upper Band Edge (nm)
Panchromatic	450	625	800
B1: Coastal Blue	400	425	450
B2: Blue	450	480	510
B3: Green	510	545	580
B4: Yellow	585	605	625
B5: Red	630	660	690
B6: Red Edge	705	725	745
B7: NIR 1	770	832.5	895
B8: NIR 2	860	950	1040

Table 2. WorldView-2 spectral bands [Data available at Dig10]. These ranges are on guidance level and, in some cases, a more detailed analysis on the relative spectral radiance response of the different bands is needed [See UC10].

we will work with TOA radiance, and other factors that may affect the obtained radiance (Earth-Sun distance, solar zenith angle, topography, bi-directional reflectance distribution function and atmospheric effects) will be ignored for simple radiometric balancing. A further analysis considering these effects to determine the surface reflectance will be carried out in future steps of the project by considering, for example, advanced radiative transfer models like 6S and the dark object subtraction technique (DOS) to remove the effects of the atmosphere [Mar+12], or consider more empirical approaches[Sta+12]. Henceforth, from now on we will stick to an educated pilot test by using TOA.

3.1.4 Almond trees digitisation

Subsequently, the perimeter of the tree's top of a sample of 400 almond trees (200 positive and 200 negative in the *Xylella fastidiosa* test) were digitised by photointerpretation. The cartographic base used has been the panchromatic band of the WorldView-2 satellite image, as it has a higher spatial resolution than the multispectral bands (40 cm pixel size), therefore allowing greater precision in the digitization process (see Figure 15).

Through a rasterization process, the tree's top perimeters, in the vector layer of polygons format, have been used to isolate the pixels of the satellite image that make up the different almond trees in the sample of 400 units, in addition to assigning the corresponding attribute (positive or negative) of the *Xylella fastidiosa* test. In the case of pixels that do not belong to almond trees, they will be labelled as agricultural land. In Figure 16 we can see the top of the almond trees digitized on the satellite image in a RGB composition of three multispectral bands B7-B5-B3 with spatial resolution of 1.6 m. The detail map allows observing the set of pixels that make up the different almond trees.

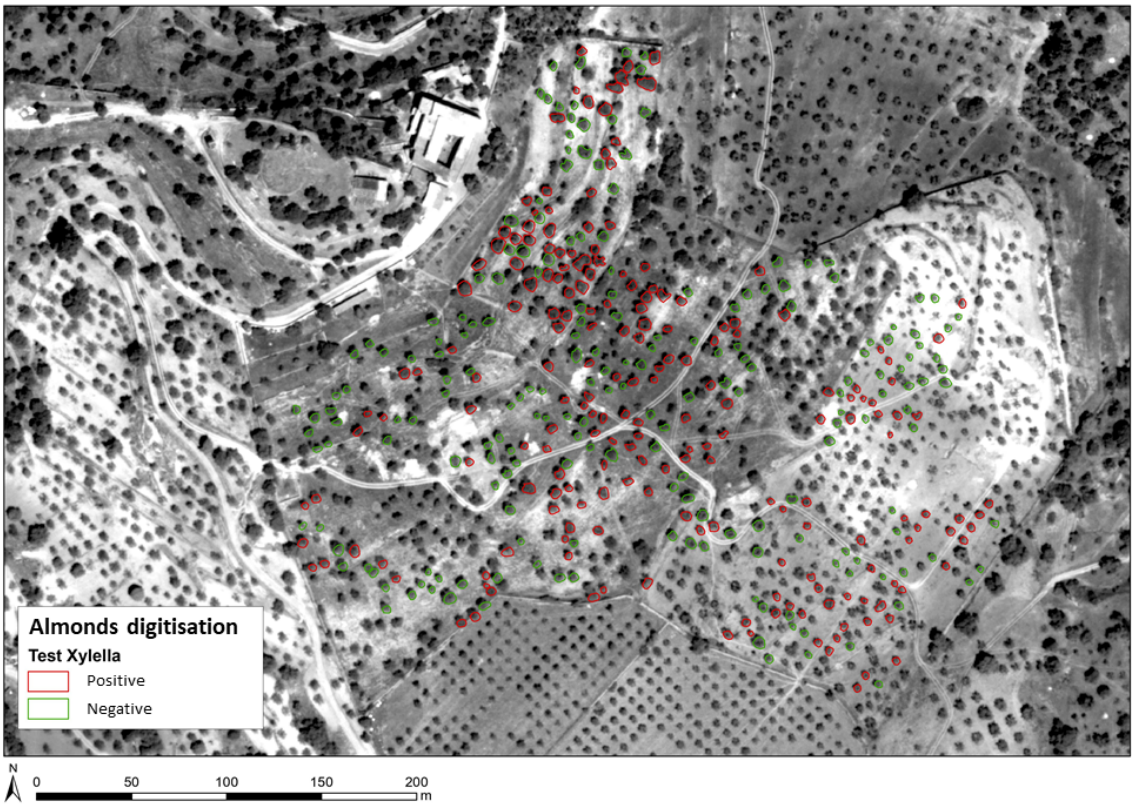


Figure 15. Digitization by photointerpretation of almond trees based on the *Xylella fastidiosa* test results according to the dataset from the Conselleria d'Agricultura. Cartographic basis: WorldView-2 satellite panchromatic band on 06/22/2011. Pixel size: 0.4m. UTM Projection, UTM zone 31N, Datum WGS1984.

In addition to the eight bands of TOA spectral radiance of the satellite, a selection of normalized indices have been calculated[3]: NDVI, GNDVI, SAVI, NPQI, CLR, CLG, BNDVI and CTR1 to have a total of 18 combinations that the neural network can incorporate into the machine learning process.

- **Normalized Difference Vegetation Index (NDVI).** This index is based on the radiometric behavior of the vegetation, related to photosynthetic activity and leaf structure of plants, allowing to distinguish between vegetation and the brightness produced by the soil. It is determined from the visible red (Red) and near-infrared light (NIR) as

$$NDVI = \frac{NIR - Red}{NIR + Red}. \tag{3.2}$$

  Particularly, it is expected that healthy trees absorb most of the visible light they receive, while diseased trees will reflect similar red and near-infrared light. This happens because in the visible part of the spectrum the leaf pigments absorb most of the energy they receive and, in the NIR, the cell walls of the leaves, which are full of

[3] Vegetation indices should be calculated with the surface reflectance of the plant but an educated approach of considering the top-of-atmosphere radiance is valid. For further discussion check Appendix 5.

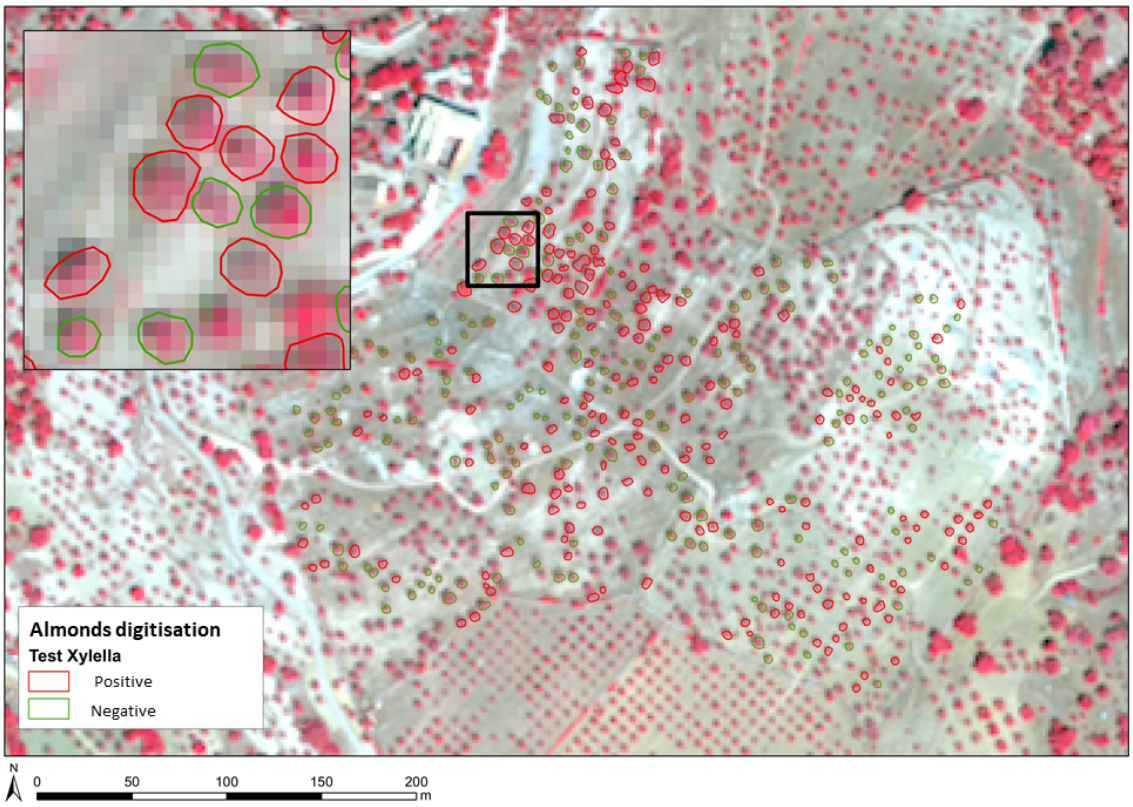


Figure 16. Digitization by photointerpretation of almond trees based on the *Xylella fastidiosa* test results according to the dataset from the Conselleria d'Agricultura. Cartographic basis: Worldview-2 satellite multispectral image on 06/22/2011 RGB: near infrared - red - green. Pixel size: 1.6m. UTM Projection, UTM zone 31N, Datum WGS1984.

water, reflect the greatest amount of energy. This no longer holds for diseased trees as they suffer from water stress, lowering the NIR reflectivity and enhancing the red band [Rou74]. In our study we will determine 2 different NDVI indices by using the 2 near-infrared channels.

- **Green Normalized Difference Vegetation Index (GNDVI).** This index is a variation of NDVI which uses the green band instead of the red band as

$$GNDVI = \frac{NIR - Green}{NIR + Green}. \tag{3.3}$$

On account of this variation, the GNDVI is more sensitive to the variation of chlorophyll in the crops than the NDVI and presents a higher saturation point [Kem+07]. In our study we will determine 2 different GNDVI indices by using the 2 near-infrared channels.

- **Soil Adjusted Vegetation Index (SAVI).** This index is also a modification of NDVI useful in areas where vegetative cover is low and the soil surface is exposed. In this situation, the soil reflectance of light in the red and near-infrared spectra can influence NDVI, hence we require the correction

$$SAVI = \frac{NIR - Red}{NIR + Red + L}(1 + L), \tag{3.4}$$

where L is the soil brightness correction factor, being $L = 0$ in very high vegetation regions[4] and $L = 1$ for areas with no green vegetation [Hue88]. From Figure 13 we observe that the Son Cotoner d'Avall farm presents low vegetation density and a large soil exposition, thus we will consider 6 different indices by taking $L \in \{0.25, 0.5, 1\}$ and the 2 near-infrared channels.

- **Normalised Phaeophytinization Index (NPQI).** This spectral index is particularly sensitive to chlorophyll degradation into phaeophytine, and has proven to be useful in X. fastidiosa detection[Pob+20]. It uses the combination of shortest bands of the visible spectrum[5]

$$NPQI = \frac{CoastalBlue - Blue}{CoastalBlue + Blue + L}(1 + L), \tag{3.5}$$

where again L is the soil brightness correction factor [Peñ+95]. Again, we will consider 4 different indices by taking $L \in \{0, 0.25, 0.5, 0.75, 1\}$.

[4] Notice that in this particular case we have NDVI = SAVI.

[5] The NPQI was originally calculated using the $415nm$ and $430nm$ bands [Bar+92]. However, as we are using the WorldView2 sensors, we determine them using the coastal blue and blue bands.

- **Chlorophyll Index Red edge (CLR)**. The chlorophyll indices are used to determine the total chlorophyll content of the leaves, as they are sensitive to its small variations and maintains its consistency across most species. This family of indices appear by establishing linear relations with the NIR band. When the red-edge band is considered, the index responds to

$$CLR = \frac{NIR1}{RedEdge} - 1. \tag{3.6}$$

- **Chlorophyll Index Green (CLG)**. As in the previous case, if we now consider the green band we have the expression [Pob+20]

$$CLG = \frac{NIR1}{Green}. \tag{3.7}$$

  CLR and CLG have both been employed in agriculture, hyperspectral remote sensing and in the detection of chlorophyll, with satisfying results in the remote sensing of X. fastidiosa [Pob+20].

- **Blue Normalized Difference Vegetation Index (BNDVI)**. This index is quite useful in areas sensitive to chlorophyll content when there is no availability of a visible blue spectral band. It is calculated from the NIR1 and blue bands as

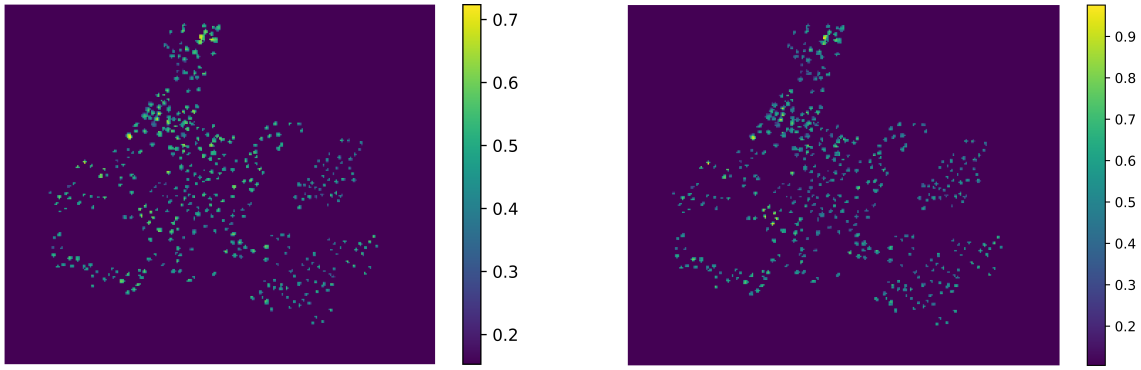
$$BNDVI = \frac{NIR1 - Blue}{NIR1 + Blue}. \tag{3.8}$$

  The BNDVI obtained from onfield robot inspection has proven to be quite useful in the detection of X. fastidiosa [Rey+19].

- **Carter Index 1 (CTR1)**. Variations in the amount of water produces changes in leaf internal structure. These alterations influence spectral reflectance in the visible red band, as well as the shortwave infrared regions. This effect is captured by the Carter Index 1 according to the expression [Pob+20]

$$CTR1 = \frac{Red}{CoastalBlue}. \tag{3.9}$$

Note that we expect the machine learning algorithms to be able to reconstruct these indices due their non-linearity power. Consequently if a large dataset is available no index should be determined. However, as we have a really small dataset, we can speed up the convergence of the algorithms by feeding them with the abovementioned indices directly, having then 26 features (8 bands and 18 indices) for each pixel. In Figure 17 we can



(a) NDVI **(b)** CLG

Figure 17. Normalized Difference Vegetation Index (NDVI) and Chlorophyll Index Green (CLG) for the 400 almond trees of the Son Cotoner d'Avall farm.

observe the NDVI and CLG indices for the set of 400 almond trees.

3.2 Data preprocessing

As it was mentioned in the previous section, our dataset consists of 8 spectral bands and 18 normalized indices obtained from the aforenamed spectral bands, having therefore 26 features for each pixel. Besides, the dataset is constituted by 400 analyzed trees with an unequal distribution of pixels among them (see Figure 18), resulting in a total of 2,316 pixels.

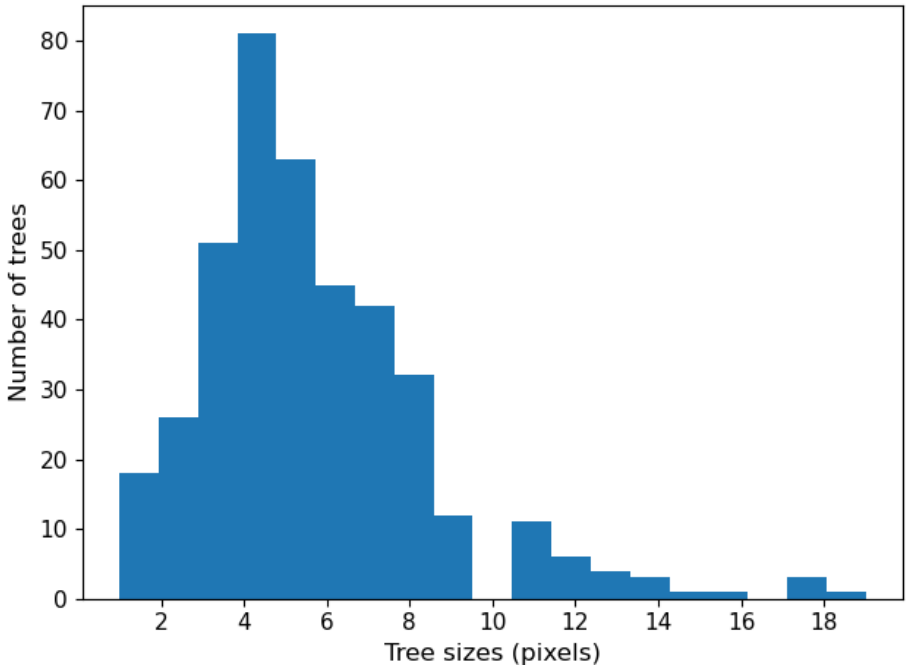


Figure 18. Histogram of tree sizes in pixels. The most common trees present a 4-pixel size, however this is not always the case as we have 1-pixel trees and largest trees with up to 19 pixels.

3.2.1 Standardization

The standardization process consist in removing the mean and scaling to unit variance the different features characterizing a problem. Hence, for a given feature x of the training set, the standard score is given by

$$z = (x - u)/s, \tag{3.10}$$

where u is the mean value accross the training samples and s is the standard deviation of the training samples. This procedure of applying individually standardization on each feature is a must do for a correct behaviour of most machine learning algorithms. Convergence of machine learning architectures suffer when the individual features are not distributed standard normally (0 mean and unit variance) [Bha20].

3.2.2 Principal Component Analysis (PCA)

Principal Component Analysis (PCA) constitutes a fast unsupervised algorithm for dimensionality reduction in datasets. It is defined as an orthogonal linear transformation which converts the data to a new coordinate system where the greatest variance of the data relies on the first coordinate, and so on [TSK16]. So, consider the dataset $\{x^{(i)}\}$, with $x^{(i)} \in \mathbb{R}^n$ representing tree pixel i along with its $n = 26$ features. In order to apply the PCA method, we require the dataset to be standardize (See section 3.2.1). To obtain the principal components we firstly compute the covariance matrix, Σ, given by [Smi02]

$$\Sigma = \frac{1}{m-1}\sum_{i=1}^{n}(x^{(i)})(x^{(i)})^T = \frac{1}{m-1}X^T X, \tag{3.11}$$

where $m = 2,316$ represents the total number of pixels and

$$X = \begin{bmatrix} \dots & x^{(1)^T} & \dots \\ & \vdots & \\ \dots & x^{(m)^T} & \dots \end{bmatrix} \in \mathbb{R}^{m\times n}. \tag{3.12}$$

At this point we can determine the variance of each feature (represented in the columns) and add them up to obtain the *total variance*. So, if we divide each feature variance by the total variance we will see how much variance each feature explains. Besides, the eigenvectors and eigenvalues of matrix Σ must be calculated (p.e. with a singular value

decomposition (SVD)) in order to have

$$V^{-1}\Sigma V = D \tag{3.13}$$

being $V \in \mathbb{R}^{nxn}$ the eigenvectors and $D \in \mathbb{R}^{nxn}$ the diagonal matrix of the covariance matrix eigenvalues

$$D_{kl} = \lambda_k \delta_{k,l}. \tag{3.14}$$

In accordance with this decomposition, the dataset can be rewritten in the new feature basis as

$$Z = V^T X \tag{3.15}$$

Now, we sort the columns of the eigenvector matrix V and eigenvalue matrix D in order of decreasing eigenvalue. This set of eigenvectors represent a new set of features (the principal components) which represent the same amount of information as the original variables. Furthermore, the total variance remains unaltered but is now distributed such that the first feature explains the most variance a single variable can explain, and so on. Particularly, the variance of each new feature k is represented by its corresponding eigenvalue, λ_k. Hence, the cumulative explained variance, g_j, by the first j principal components will be

$$g_j = \sum_{k=1}^{j} \lambda_k \tag{3.16}$$

Coming back to our particular problem, a PCA may be applied in order to reduce the dimensionality of the system and get rid of redundant features which provide nearly no information. The results obtained from the PCA are shown in Figure 19. There we can observe that approximately 93% of the problem variance can be explained with just the first 2 principal components and above 99% with just the 5 first principal components. This reveals that many of the calculated indices may present correlation, providing each of them scarce new information with respect to the others. However, given the small number of features we have and the reduced dataset size, we will preserve all the features in the subsequent analysis. Otherwise, in further steps of the project when larger datasets are available a PCA may be a must do.

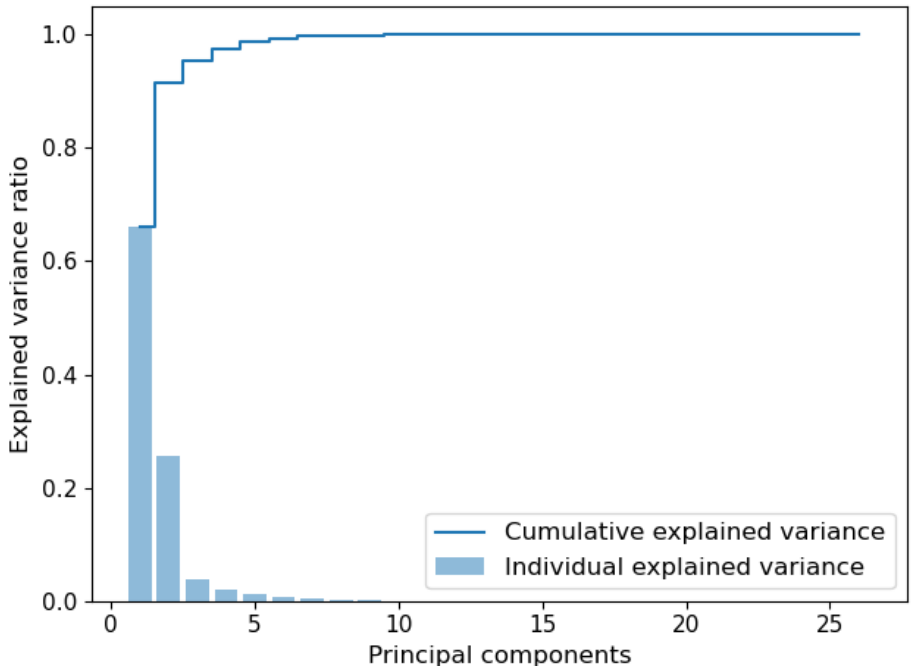


Figure 19. Principal component analysis to the sample of almonds pixels. Observe that approximately 93% of the features variance can be explained with just the two first principal components.

3.2.3 Training set vs Test set

Once the preprocessing stage is complete we end up with a standardize dataset consisting of 26 features (8 spectral bands and 18 vegetation indices) and a low number of analyzed trees (<1000) each of them with its respective pixels. Now, we must split our dataset as follows:

- *Training set (75 % of the dataset).* The sample of data used to fit the different machine learning models. The common practice is to standardize it and then fit the cross-validation set to this transformation (with the mean and standard deviation of the training set).
- *Cross-validation set (25 % of the dataset).* The sample of data used to provide an unbiased evaluation of a model fit on the training set while tuning the model hyperparameters. Notice that the estimation would become biased as the model starts to get skill on the cross-validation set.

A common and useful practice to get an unbiased estimation of the model performance consist in creating also a *test dataset* [Sha17]. However, as we do not have a large dataset we do not consider having this testing set by the moment, so we will use a k-Fold Cross-validation instead. A common practice while training machine learning algorithms is to plot the cost function evaluated over the training and testing set in order to prevent our

model from suffering underfitting or overfitting.

3.2.4 k-Fold Cross-Validation

This procedure allows to obtain an unbiased measure of the models performance. In this procedure, we firstly divide the dataset into k folds and, for a given model, each one of the k folds takes turn to be the hold-out cross-validation set and the other k-1 folds are used as training set. The model performance is tested over the hold-out set and the overall performance is taken to be the average of the performances on all the k folds. A scheme is shown Figure 20.

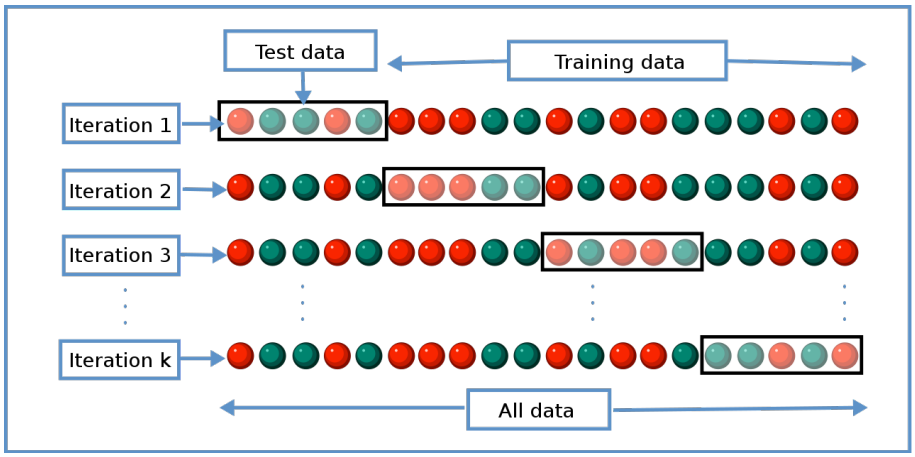


Figure 20. Diagram of k-fold cross-validation [Guf20].

Cross-validation turns out to be quite useful in problems where the dataset size is so small that one can not afford to hold out part of the data just for testing purposes [Zhe15].

3.3 Machine Learning models

3.3.1 SVM

As we mentioned in Section 3.3.1, we are using a Gaussian SVM algorithm due to its proven good performance at *Xylella fastidiosa* remote sensing [Pob+20]. In particular, we use a 7-fold cross-validation over the set of 400 almond trees where we apply the assumption that each pixel can represent an individual tree, having therefore a dataset size of 2,316 samples. For each evaluation of the model over the hold-out set the prediction will be positive if the output probability is larger than 0.5, and negative otherwise.

The Gaussian SVM classifier presents two main parameters: $\gamma = 1/2\sigma^2$ and C, where C represents the penalty for misclassifying a data point in the optimization . Consequently,

γ measures the spread of the decision region, being broader as lower is γ and viceversa. Moreover, when C is small the classifier presents high bias, and high variance when C is large. After a naive search, the parameters that gave the best k-fold performance where $\gamma = 0.0193$ and $C = 100$.

3.3.2 ANN: 1 pixel - 1 tree

Just as in the case of SVM, we take the assumption that each pixel represents an individual tree. Again, we apply a 7-fold cross-validation and after a naive search we take as the both best and simplest ANN architecture the one shown in Figure 21 which presents the following characteristics:

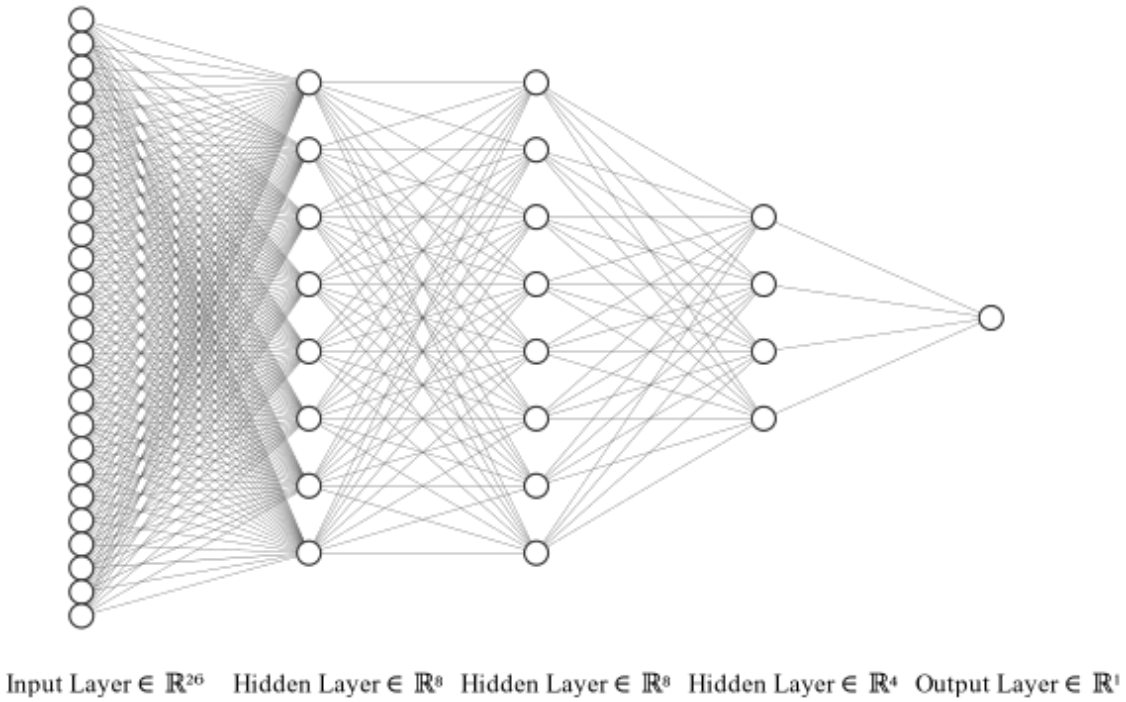


Figure 21. Artificial Neural Network diagram.

- *First Hidden Layer*. This first hidden layer is formed by 8 neurons with ReLU activation functions and both L2 regularization ($\lambda = 0.0006$) and dropout ($p^{[1]} = 0.2$).
- *Second Hidden Layer*. The second hidden layer also has 8 nodes with ReLU activation function and a L2 regularization ($\lambda = 0.0006$).
- *Third Hidden Layer*. The third hidden layer presents 4 neurons with ReLU activation function and a L2 regularization ($\lambda = 0.0003$).
- *Output Layer*. The output layer presents just one node with a sigmoid activation function for binary classification.

Hence, the ANN is composed of 329 trainable parameters which are actualized using an Adam optimizer[6] throughout the 2500 **epochs** (evaluations of the whole training set)[7].

3.3.3 ANN: Average over pixels

At this point one important detail come into light: the disease may spread in a non uniform way through the almond trees canopy (see Figure 22). Hence, some pixels may show a healthy pattern for an infected tree and, consequently the crucial information to detect *Xylella fastidiosa* could be condensed in a couple of pixels.

Figure 22. Uneven distribution of *Xylella fastidiosa* symptoms over an olive tree [Uni18].

With this idea in mind, we divide the dataset into a training set of 300 trees and a testing set of 100 trees. With the training set we fit the same architecture mentioned in Section 3.3.2 using a 5-fold cross-validation, Adam optimizer and 3000 epochs. Afterwards, we take the model with the best performance and follow the subsequent guidelines:

1. Pick up one tree from the 100 size cross-validation set.
2. For each pixel constituting the tree use the trained model to estimate its probability of being infected.
3. Average over the individual pixels probabilities.
4. If the average is greater than 0.5 we predict the tree to be infected, and healthy in the opposite case.

[6]For a detailed explanation about Adam optimizer check: https://towardsdatascience.com/adam-latest-trends-in-deep-learning-optimization-6be9a291375c

[7]All the neural networks shown in this project have been programmed in the Keras-Tensorflow deep learning framework.

ANN: Average over pixels with clipping

This model has the same structure and follows the same training as the previous one with the slight difference:

4. If the average is greater than 0.5 or one of the tree pixels presents a probability of being infected greater than 0.75, then we predict the tree to be infected, and healthy in the opposite case.

With this modification we try to deal with the uneven *Xylella fastidiosa* distribution in a tree by considering that a "clearly" infected branch is enough to classify the tree as infected.

3.3.4 Simple LSTM RNN

Exploiting the idea that *Xylella fastidiosa* symptons may manifest in an uneven way through the canopy pixels we will now train a simple LSTM RNN. However, this kind of architectures require from large training sets due to the large amount of parameters they present. Hence, we are just going to present mainly a proof of concept. With this in mind, we divide the dataset into a training set of $m_{train} = 300$ trees and a testing set of $m_{test} = 100$ trees. Then, we write the training set as a matrix $\mathbf{X}$ with dimensions (m_{train}, L_{max}, n), being $L_{max} = 19$ the largest tree size and $n = 26$ the number of features. Those rows and columns of trees with a size smaller than 19 pixels will be padded with 0's. Then, the architecture which showed a better performance is the following:

- *LSTM layer*. This first hidden layer presents a LSTM layer with 8 units (dimensionality of the output space) with ReLU activation functions and both L2 regularization ($\lambda = 0.0003$).
- *Dense layer*. The second hidden layer is a dense layer[8] with 8 nodes, ReLU activation function and a L2 regularization ($\lambda = 0.2$) and dropout ($p^{[1]} = 0.1$)
- *Output Layer*. The output layer presents just one node with a sigmoid activation function for binary classification.

Thus, the simple LSTM RNN is formed by 1,201 trainable parameters which are updated using the Adam optimizer through 100 epochs. The reduction in the number of epochs with respect to ANN architectures previously detailed relies on the increase of the number of trainable parameters which makes the neural network more prone to overfit the training

[8] When using deep learning architectures different from ANN, simple layers like the ones of an ANN are called *dense layer*.

set and offer a worse performance over the testing set. This technique of reducing the number of epochs up to the optimal point is known as *early stopping*.

3.3.5 Bidirectional LSTM RNN

This architecture enhances the information transmission forward and backward in the neural network, so it is expected that offers a better performance over our particular problem. The procedure followed during the training of the network is nearly the same as the one explained for the simple LSTM RNN. The main differences lay in the architecture which we present hereunder:

- *LSTM bidirectional layer*. This first hidden layer presents a forward and backward LSTM layers with 8 units each. Both of them present a ReLU activation function for the output.
- *Dense layer*. The second hidden layer is a dense layer with 8 nodes, ReLU activation function and a L2 regularization ($\lambda = 0.2$) and dropout ($p^{[1]} = 0.2$)
- *Output Layer*. The output layer presents just one node with a sigmoid activation function for binary classification.

Consequently, the bidirectional LSTM RNN is formed by 2,385 trainable parameters which are updated using the Adam optimizer through 50 epochs. We are again reducing the number of epochs to avoid overfitting.

3.3.6 Project workflow

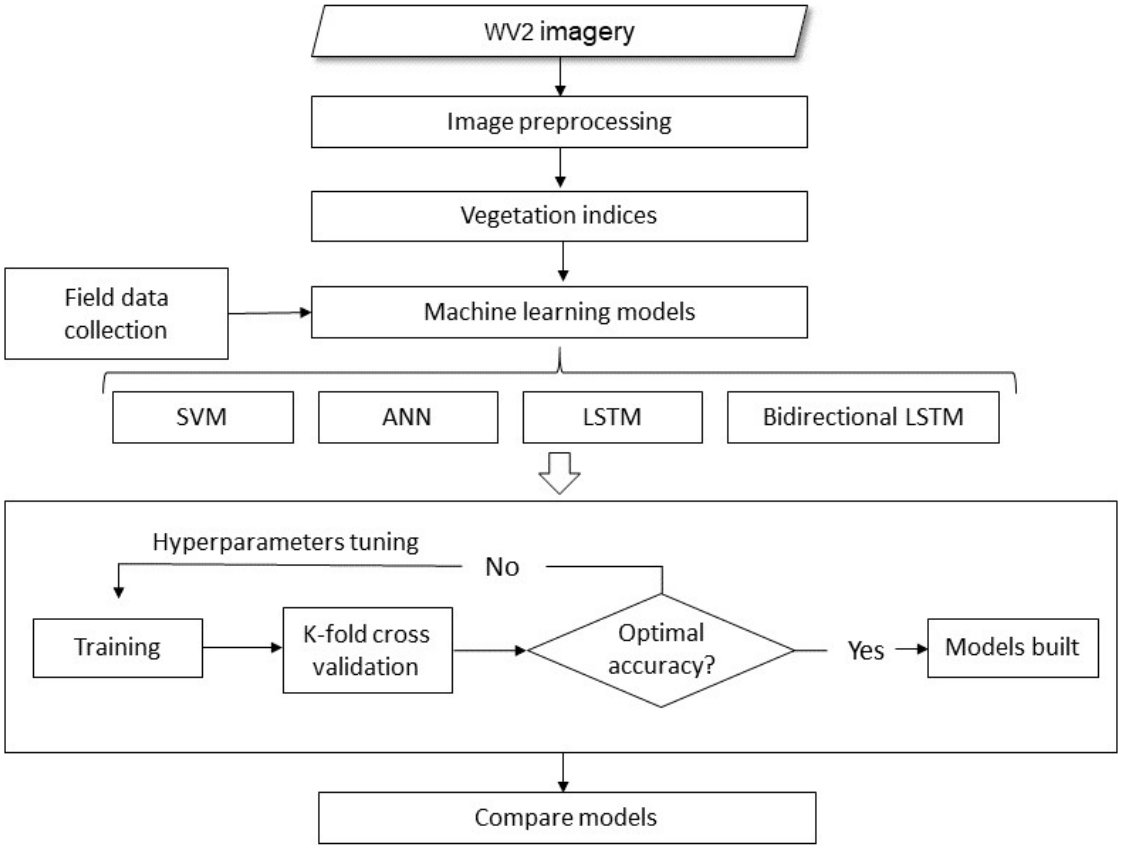


Figure 23. Workflow for remote sensing of *Xylella fastidiosa* using WV2 imagery and machine learning techniques and field study data.

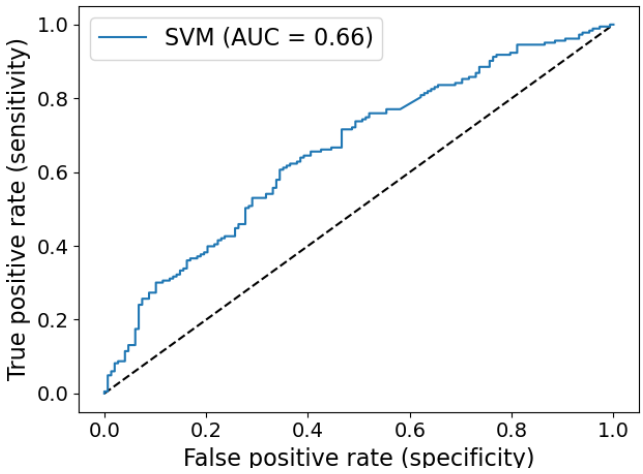
4. Results

Abstract

In this chapter we present the different results obtained and the performances achieved by the models. Hereunder, a comparison between the different models capabilities is established and a discussion on which future steps may be taken is done. Finally, some hopeful results related to the dataset characteristics are highlighted.

4.1 SVM

With the configuration described in Section 3.3.1, the performance achieved by the SVM classifier presents some strengths and weaknesses. Particularly, the algorithm proves to be efficient in the task of identifying many positive examples (see Table 3) which results in a large recall value of 0.814. Indeed, the model overestimates the number of positive cases, presenting a precision of 0.649 (see Table 9). In any case, the 0.8 accuracy achieved using SVM in remote sensing in Apulia [Pob+20] was far from uncontested as we reached a 0.59 ± 0.02 accuracy. The reasons of this discrepancy relies on the different imagery source (aerial imagery, hyperspectral cameras and thermal cameras vs satellite imagery) and the different sizes of the datasets ($7,000$ olives vs 400 almonds) among others like the large temporal period between the satellite image and the q-PCR tests, which will be detailed in Section 5.



		Model prediction	
		Positive	Negative
Label	Positive	157	36
	Negative	85	53

Table 3. SVM ROC curve and confusion matrix over the fold with largest accuracy.

4.2 Artificial Neural Network: 1 pixel - 1 tree

The ANN architecture employed show some interesting results. Firstly, the algorithm is capable of identifying the positive examples with a recall of 0.765 and an accuracy of 0.61 ± 0.02. In general terms, the performance of the ANN is quite similar to the one achieved by the SVM, with a slightly better precision and accuracy. However, we expect it to be more powerful in future steps of the project when a larger dataset is available.

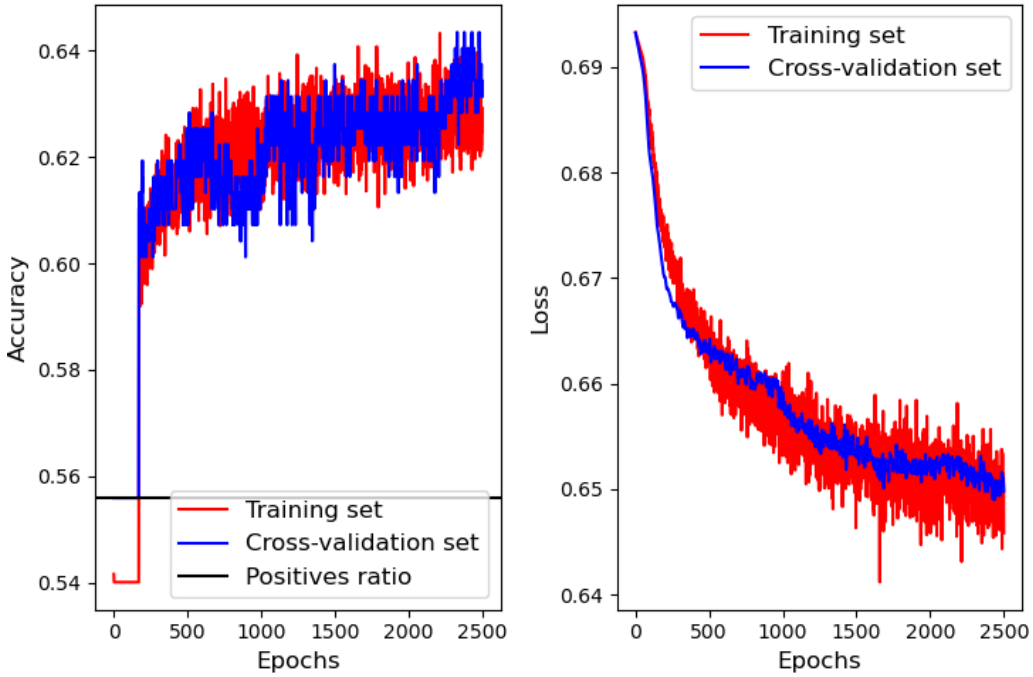
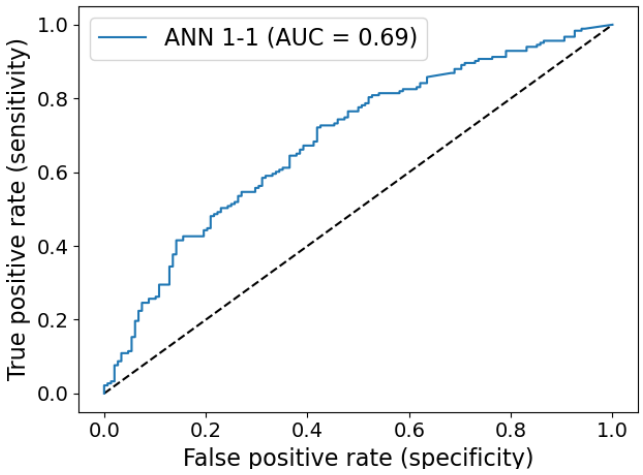


Figure 24. Learning curves for the ANN 1 pixel - 1 tree architecture. It shows the accuracies as well as the ratio of positive examples (left) and the cost functions (right). Both quantities are evaluated over the training and cross-validation sets for every epoch.



		Model prediction	
		Positive	Negative
Label	Positive	140	43
	Negative	71	77

Table 4. ANN 1-1 ROC curve and confusion matrix over the fold with largest accuracy.

4.3 Artificial Neural Network: Average over pixels

Things improve when we introduce the method of averaging over the pixels of a given tree. We have seen that the performance of the ANN 1-1 architecture is weak due to the poor quality of the dataset and to the uneven distribution of *Xylella fastidiosa* infection all over the canopy. Despite this limited power, the ANN 1-1 architecture performance is enhanced when the averaging trick is considered. Particularly, we can observe an increase of ≈ 0.05 in accuracy and of ≈ 0.12 in recall (see Table 6). Otherwise, the clipping tool shows no performance improvement but it should not be ruled out and further analysis is required.

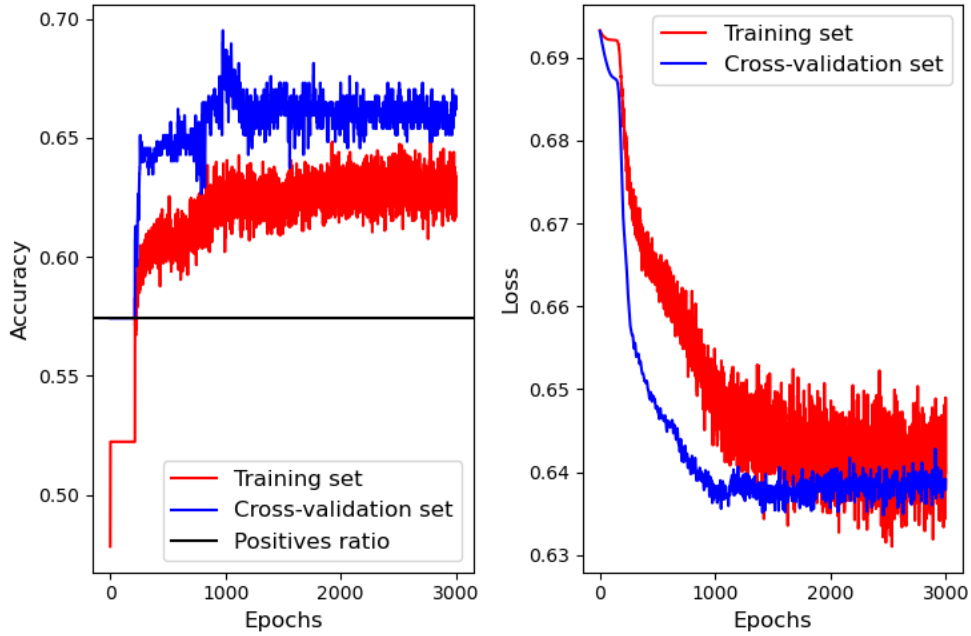


Figure 25. Learning curves for the ANN 1 pixel-1 tree used for average classification. It shows the accuracies and the ratio of positive examples (left), and the cost functions (right). Both quantities are evaluated over the training and cross-validation sets for every epoch.

		Model prediction	
		Positive	Negative
Label	Positive	51	10
	Negative	20	19

		Model prediction	
		Positive	Negative
Label	Positive	51	10
	Negative	21	18

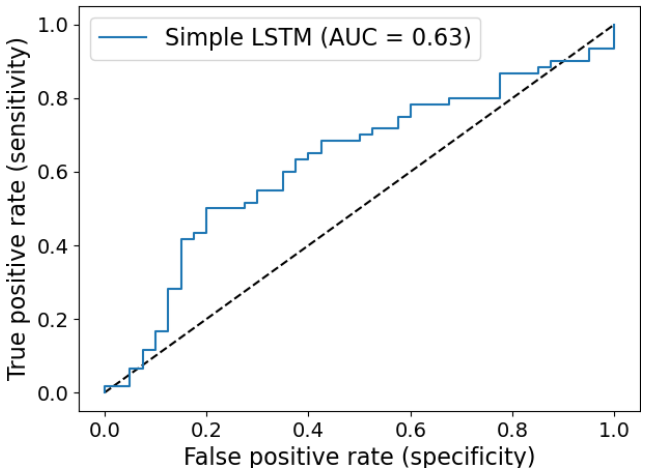
Table 5. ROC curve and confusion matrix for ANN with pixels averaging with and without clipping.

Metric	ANN 1-1	ANN Average	ANN Average + clipping
Accuracy	0.654	0.700	0.690
AUC	0.670	0.670	0.660
Recall	0.718	0.836	0.836
Precision	0.691	0.718	0.708
F1-Score	0.704	0.773	0.767
k-fold accuracy	0.63 ± 0.03	-	-
k-fold cost	0.66 ± 0.03	-	-

Table 6. Performance of the different ANN models. The ANN 1-1 column shows the performance of this model over the fold with the largest accuracy concerning the 300 trees set. The other two columns show the performance of the ANN 1-1 model applied to the pixels of each tree and averaging the probabilities obtained for the 100 trees set.

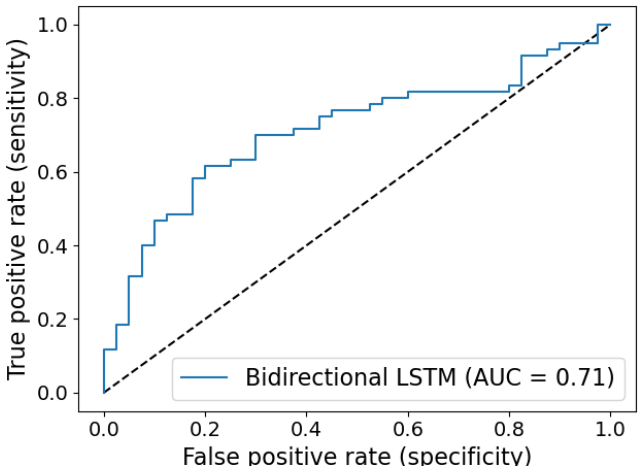
4.4 Simple and bidirectional LSTM RNN

These two deep learning architectures are the most sophisticated algorithms among them all. The results they present are promising, in special the bidirectional LSTM RNN which shows a 0.69 accuracy in spite of the lack of data required to train these deep architectures (see Table 9). However, these results may be biased as we are not performing neither a k-fold cross validation nor having a testing set. Consequently, this represents an encouraging proof of concept that definitely must be explored further on.



		Model prediction	
		Positive	Negative
Label	Positive	45	15
	Negative	21	19

Table 7. ROC curve and confusion matrix for simple LSTM RNN.



		Model prediction	
		Positive	Negative
Label	Positive	41	19
	Negative	12	28

Table 8. ROC curve and confusion matrix for bidirectional LSTM RNN.

4.5 Comparison of models

The performance of the models is limited with accuracies ranging from 0.6 to 0.7 . This cannot be seen as a defeat, however, it settles an optimistic basis for future work if we account for the limited size and quality of the dataset. An interesting point is that nearly all the models present a large recall compared with the precision. This means that they all have a special capability for detecting positive examples. Indeed, this is an important result as we want our models to prevent infected almond trees to slip away without been detected. Recall that strong countermeasures are being taken to control the expansion of the epidemic. Nevertheless, this has not happened by chance. The actual dataset is theoretically balanced (200 positives and 200 negatives) according to the q-PCR tests taken in 2018. However, the satellite image is dated 22 June 2011 and consequently we expect that the dataset was unbalanced at that point: more negative than positive trees at that point which has resulted in a balanced dataset due to *Xylella fastidiosa* spread. Consequently, those trees which were already infected in 2011 will most likely have developed the disease,

showing a deteriorated condition in 2018. The machine learning models have been able to detect those deteriorated trees with an incredible accuracy, though struggling with those trees which were healthy or asymptomatic in 2011 and that now are clearly infected. This fact opens the possibility of enhancing the models' performance were a 2018 satellite image be available.

Metric	SVM	ANN 1-1	ANN Average	Simple LSTM RNN	Bidirectional LSTM RNN
Accuracy	0.634	0.656	0.700	0.640	0.690
AUC	0.660	0.690	0.670	0.630	0.710
Recall	0.814	0.765	0.836	0.750	0.683
Precision	0.649	0.664	0.718	0.682	0.774
F1-Score	0.722	0.711	0.773	0.714	0.726
k-fold accuracy	0.59 ± 0.02	0.61 ± 0.02		-	
k-fold cost	0.667 ± 0.006	0.66 ± 0.01		-	

Table 9. Machine learning models performance. First five metrics are evaluated over a unique fold and the last two metrics are averaged cross evaluation results (more unbiased metrics).

5. Conclusions and future steps

To sum up, we have started this project by presenting a real concern which affects the agriculture and economy of the Balearic Islands: the *Xylella fastidiosa* epidemic. A detailed analysis of its characteristics was made by reviewing the related literature along with a field study of the most recent techniques in detection *Xylella fastidiosa*. Hereunder, a cooperative scheme between satellite imagery and Machine Learning algorithms is presented. Afterwards, a revision of some relevant Machine Learning algorithms and classification metrics is made. Then, the results of the q-PCR tests supplied by the Conselleria are presented along with the satellite image, and a detailed study of the data is performed. Finally, different Machine Learning algorithms are trained and the different results obtained are compared between the models.

The goal of the this project, based on satellite images analysis and infield *Xylella fastidiosa* test, was to develop an algorithm capable of identifying which almond trees may be infected. Particularly, we expected remote sensing to be more accurate than a visual inspection by plant pathologists with regard to early detect *Xylella fastidiosa* infection. However, the results obtained are still prohibitively inaccurate for practical applicability due to the different faced shortcomings. Specifically, the main penalty to this project has been the large temporal period between the satellite image (2011) and the q-PCR tests performed (2018). This may have affected the quality of the dataset by introducing wrong labels: trees that are infected in 2018 may have been healthy when the satellite image was taken in 2011 (just see how fast the epidemic has evolved in Figure 1). In the same way, the limited size of the dataset has penalized both the performance of the trained models and the possibility of exploring deeper neural network architectures. Nevertheless, as a proof of concept, this project has shown promising results as the trained models have been capable of detecting infected trees up to a certain accuracy. Consequently, it leads the way to promising improvements when fresher satellite imagery is available.

Future work of this project will be based on repeating the same procedure with a satellite image closer to 2018 as well as expanding the dataset size by considering the whole set of almond trees in the Son Co-toner d'Avall farm. Moreover, higher quality imagery from the newer WorldView3 satellite could be employed. Accordingly, the process of trees digitization should be automatized which may be done using new Machine Learning algorithms or more traditional remote sensing methods. We expect this improvements to enhance the performance of the Machine Learning models allowing, for example, to study the spatio-temporal evolution of the epidemic if satellite imagery from different periods of time is provided. In this sense, new Machine Learning architectures like Recurrent Neural Networks with attention and transformers would be practicable thanks to the expanded dataset. Besides, future steps might explore the incorporation of higher spatial- and temporal-resolution imagery, as well as transferring knowledge from deep learning models trained on related tasks like crop yield monitoring. In conclusion, the encouraging results obtained in this project along with the chance of great future improvements, enable the possibility of practical application.

Bibliography

Main sources

[Agg18] Charu C Aggarwal. "An introduction to neural networks". In: *Neural Networks and Deep Learning*. Springer, 2018, pp. 1–52.

[Agu+13] M Aguilar et al. "Radiometric comparison between GeoEye-1 and WorldView-2 panchromatic and multispectral imagery". In: Congress INGEGRAF-ADM-AIP PRIMECA Madrid, Spain; 2013.

[Alm16] Rodrigo PP Almeida. "Can Apulia's olive trees be saved?" In: *Science* 353.6297 (2016), pp. 346–348.

[Alm18] Rodrigo PP Almeida. "Emerging plant disease epidemics: Biological research is key but not enough". In: *PLoS biology* 16.8 (2018), e2007020.

[Aut+15] EFS Authority et al. "Scientific opinion on the risks to plant health posed by Xylella fastidiosa in the EU territory, with the identification and evaluation of risk reduction options." In: *Efsa Journal* 13.1 (2015).

[Bar+92] Jeremy D Barnes et al. "A reappraisal of the use of DMSO for the extraction and determination of chlorophylls a and b in lichens and higher plants". In: *Environmental and Experimental botany* 32.2 (1992), pp. 85–100.

[Dig10] DigitalGlobe. "The Benefits of the Eight Spectral Bands of WorldView-2". In: *White Paper* (2010).

[Hor+20] A Hornero et al. "Monitoring the incidence of Xylella fastidiosa infection in olive orchards using ground-based evaluations, airborne imaging spectroscopy and Sentinel-2 time series through 3-D radiative transfer modelling". In: *Remote Sensing of Environment* 236 (2020), p. 111480.

[Jak06] Vikramaditya Jakkula. "Tutorial on support vector machine (svm)". In: *School of EECS, Washington State University* 37 (2006).

[Kem+07] A Kemerer et al. "Comparación de índices espectrales para la predicción del IAF en canopeos de maíz". In: *XII Congreso de Teledetección*. 2007.

[Lia+18] Konstantinos G Liakos et al. "Machine learning in agriculture: A review". In: *Sensors* 18.8 (2018), p. 2674.

[Mar+12] J Martin et al. “Atmospheric correction models for high resolution WorldView-2 multispectral imagery: a case study in Canary Islands, Spain”. In: *Remote Sensing of Clouds and the Atmosphere XVII; and Lidar Technologies, Techniques, and Measurements for Atmospheric Remote Sensing VIII*. Vol. 8534. International Society for Optics and Photonics. 2012, 85340O.

[Mis18] Aditya Mishra. “Metrics to evaluate your machine learning algorithm”. In: *Towards Data Science* (2018).

[MP19] Y Martınez and A Palacio-Bielsa. “Estimación del impacto económico de Xylella fastidiosa en Aragón”. In: *ITEA, información técnica económica agraria: revista de la Asociación Interprofesional para el Desarrollo Agrario (AIDA)* 115.2 (2019), pp. 175–191.

[NP16] NJ Nalini and S Palanivel. “Music emotion recognition: The combined evidence of MFCC and residual phase”. In: *Egyptian Informatics Journal* 17.1 (2016), pp. 1–10.

[PEL17] Reid Pryzant, Stefano Ermon, and David Lobell. “Monitoring ethiopian wheat fungus with satellite imagery and deep feature learning”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 2017, pp. 39–47.

[Peñ+95] J Peñuelas et al. “Reflectance assessment of mite effects on apple trees”. In: *International Journal of Remote Sensing* 16.14 (1995), pp. 2727–2733.

[Per+13] Luigi Perotti et al. “Remote sensing and hydrogeological methodologies for irrigation canal leakage detection: the Osasco and Fossano test sites (Northwestern Italy)”. In: *Geophys. Res. Abstr., EGU2013-5705, EGU General Assembly* (2013).

[Pob+20] T Poblete et al. “Detection of Xylella fastidiosa infection symptoms with airborne multispectral and thermal imagery: Assessing bandset reduction performance from hyperspectral analysis”. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 162 (2020), pp. 27–40.

[Rey+19] Beatriz Rey et al. “XF-ROVIM. A field robot to detect olive trees infected by Xylella fastidiosa using proximal sensing”. In: *Remote Sensing* 11.3 (2019), p. 221.

[Rou74] JW Rouse. “Monitoring the vernal advancement of retrogradation of natural vegetation, NASA/GSFG, Type III”. In: *Final Report* 371 (1974).

[Sam59] Arthur L Samuel. “Some studies in machine learning using the game of checkers”. In: *IBM Journal of research and development* 3.3 (1959), pp. 210–229.

[Sha17] Tarang Shah. “About train, validation and test sets in machine learning”. In: *Towards Data Science* 6 (2017).

[Smi02] Lindsay I Smith. *A tutorial on principal components analysis*. Tech. rep. 2002.

[Sta+12] Grant William Staben et al. “Empirical line calibration of WorldView-2 satellite imagery to reflectance data: Using quadratic prediction equations”. In: *Remote sensing letters* 3.6 (2012), pp. 521–530.

[T+14] K Tumber, J Alston, Kate Fuller, et al. “Pierce’s disease costs California $ 104 million per year”. In: *California Agriculture* 68.1 (2014), pp. 20–29.

[TSK16] Pang-Ning Tan, Michael Steinbach, and Vipin Kumar. *Introduction to data mining*. Pearson Education India, 2016.

[UC10] Todd Updike and Chris Comp. “Radiometric use of WorldView-2 imagery”. In: *Technical Note* (2010), pp. 1–17.

[Zar+18] PJ Zarco-Tejada et al. “Previsual symptoms of Xylella fastidiosa infection revealed in spectral plant-trait alterations”. In: *Nature Plants* 4.7 (2018), pp. 432–439.

[Zhe15] Alice Zheng. “Evaluating machine learning models: a beginner’s guide to key concepts and pitfalls”. In: (2015).

Internet sources

[AA20] Afshine Amidi and Shervine Amidi. *CS 230 - Recurrent Neural Networks Cheatsheet*. https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-recurrent-neural-networks. 2020.

[Bha20] Aniruddha Bhandari. *Feature Scaling for Machine Learning: Understanding the Difference Between Normalization vs. Standardization*. https://www.analyticsvidhya.com/blog/2020/04/feature-scaling-machine-learning-normalization-standardization/. 2020.

[Cor20] Satellite Imaging Corporation. *WorldView-2 Satellite Sensor*. https://www.satimagingcorp.com/satellite-sensors/worldview-2/. 2020.

[Dik19] Atria Dika Puspita. *Get to Know How Artificial Neural Network Formed in Computer Science*. https://mc.ai/get-to-know-how-artificial-neural-network-formed-in-computer-science/. 2019.

[ESA20] European Space Agency (ESA). *WorldView-2 Objectives*. https://earth.esa.int/eogateway/missions/worldview-2/objectives. 2020.

[eur20] europapress. *La bacteria que asesina olivos y almendros*. https://www.europapress.es/illes-balears/noticia-ascienden-834-positivos-xylella-fastidiosa-baleares-20190219112531.html. 2020.

[Gle19] Stephanie Glen. *ROC Curve Explained in One Picture*. https://www.datasciencecentral.com/profiles/blogs/roc-curve-explained-in-one-picture. 2019.

[Guf20] Gufosowa. *Cross-validation (statistics)*. https://en.wikipedia.org/wiki/Cross-validation_(statistics)/media/File:K-fold_cross_validation_EN.svg. 2020.

[Hue88] AR Huete. *A Soil-Adjusted Vegetation Index (SAVI). Remote Sensing of Environment, 25, 295-309*. 1988.

[Jai17] Rashmi Jain. *Simple Tutorial on SVM and Parameter Tuning in Python and R | HackerEarth Blog*. https://www.hackerearth.com/blog/developers/simple-tutorial-svm-parameter-tuning-python-r/. 2017.

[LL01] DK Lynch and WC Livingston. *Limits of the eye's overall range of sensitivity extends from about 310 to 1050 nanometers. Cambridge, UK*. 2001.

[Mal20] Diario Mallorca. *Cifran en un millón los almendros infectados por la Xylella en Mallorca*. https://www.diariodemallorca.es/mallorca/2017/10/08/cifran-millon-almendros-infectados-xylella/1253864.html. 2020.

[Ope20] OpenCV. *Introduction to Support Vector Machines*. https://docs.opencv.org/2.4/doc/tutorials/ml/introduction_to_svm/introduction_to_svm.html. 2020.

[Uni18] Swansea University. *Eye-in-the-sky to save olive trees*. https://phys.org/news/2018-06-eye-in-the-sky-olive-trees.html. 2018.

[UU20] Scientific United Nations Educational and Cultural Organization (UNESCO). *Lesson 3. Radiometric Correction of Satellite Images: When and Why Radiometric Correction is Necessary*. https://cwcaribbean.aoml.noaa.gov/bilko/module7/lesson3/. 2020.

[Vid20] Ignacio Zafra Vidal Maté. *La bacteria que asesina olivos y almendros*. https://elpais.com/economia/2017/09/09/actualidad/1504976419_100239.html. 2020.

Appendices

Appendix 1 - Radiance vs Reflectance for Vegetation Indices

In general, it is much convenient to use reflectance at the top of canopy to prevent the different distortions from disrupting the calculations. Particularly, the solar spectral irradiance for a given image varies depending on the Earth-Sun distance and the solar zenith angle during the individual image acquisition. This variation will cause two scenes of the same area taken on different times to have different radiances. The difference can be minimized by correcting imagery for Earth-Sun distance and solar zenith angle. Furthermore, the atmospheric effect depends on the wavelength considered, affecting differently the 8 multispectral WV2 bands (see Figure 26).

However, this atmospheric effects may not be that important if we are not comparing the evolution of canopy throughout a period of a time, i.e. a sequence of WV2 images of the zone, which is a subsequent task of the project. In this case different methods can be used to correct the atmospheric effect like 6S and the dark object subtraction technique (DOS) [Mar+12], or other empirical approaches [Sta+12]. Therefore, it is perfectly valid to use TOA radiance for the determination of vegetation indices.

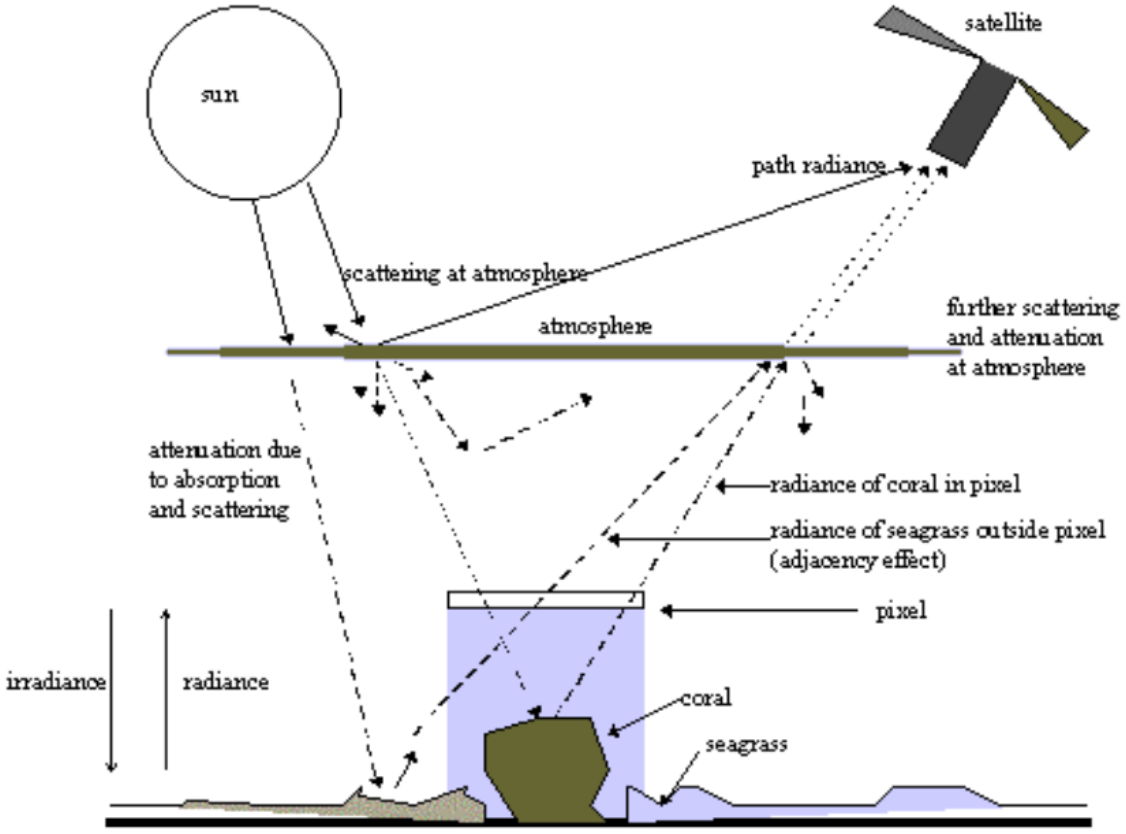


Figure 26. Scheme of the atmospheric and radiometric effects in the WV2 satellite measurements [UU20].